EXHIBIT 1

US006591153B2

(12) **United States Patent**
Crampton et al.

(10) **Patent No.: US 6,591,153 B2**
(45) **Date of Patent: Jul. 8, 2003**

(54) **SYSTEM AND METHODS FOR SCHEDULING MANUFACTURING RESOURCES**

(75) Inventors: **Myrick Crampton**, Germantown, MD (US); **George Zdravecky**, Washington, DC (US)

(73) Assignee: **Manugistics, Inc.**, Rockville, MD (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 13 days.

(21) Appl. No.: **09/974,801**

(22) Filed: **Oct. 12, 2001**

(65) **Prior Publication Data**

US 2002/0107600 A1 Aug. 8, 2002

Related U.S. Application Data

(60) Provisional application No. 60/239,280, filed on Oct. 12, 2000.

(51) **Int. Cl.**[7] **G06F 19/00**

(52) **U.S. Cl.** **700/103**; 700/99; 700/100; 700/104; 700/111; 705/8

(58) **Field of Search** 700/99, 100, 103, 700/104, 111, 101, 102; 705/8

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,744,028	A	*	5/1988	Karmarkar	705/8
5,369,570	A		11/1994	Parad	364/401
5,467,285	A	*	11/1995	Flinn et al.	700/95
5,720,157	A		2/1998	Ross	53/445
5,764,543	A		6/1998	Kennedy	364/578
5,808,891	A	*	9/1998	Lee et al.	700/108
5,835,377	A	*	11/1998	Bush	700/99
5,983,195	A		11/1999	Fierro	705/10
6,308,107	B1	*	10/2001	Conboy et al.	700/121
6,415,196	B1	*	7/2002	Crampton et al.	700/100

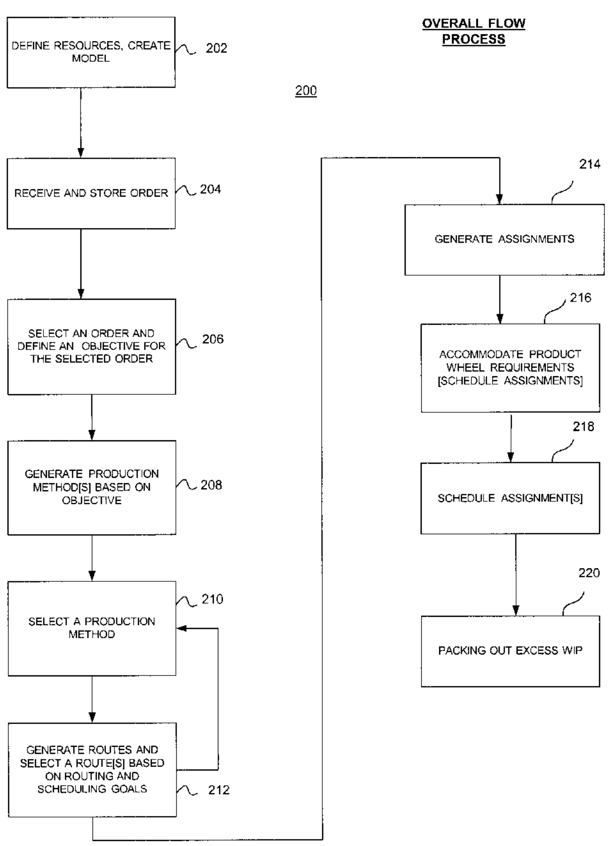
* cited by examiner

Primary Examiner—Leo Picard
Assistant Examiner—Michael D Masinick
(74) *Attorney, Agent, or Firm*—Hogan & Hartson LLP

(57) **ABSTRACT**

A system and method for scheduling manufacturing resources based on user defined scheduling and routing goals. Attributes and constraints may be assigned to manufacturing resources and SKUs that facilitates the planning and scheduling of the resources. Resource transitions may be restricted only to desirable transition by using product wheels. Excess work-in-process may be minimized by scheduling orders that consume the excess work-in-process.

64 Claims, 15 Drawing Sheets



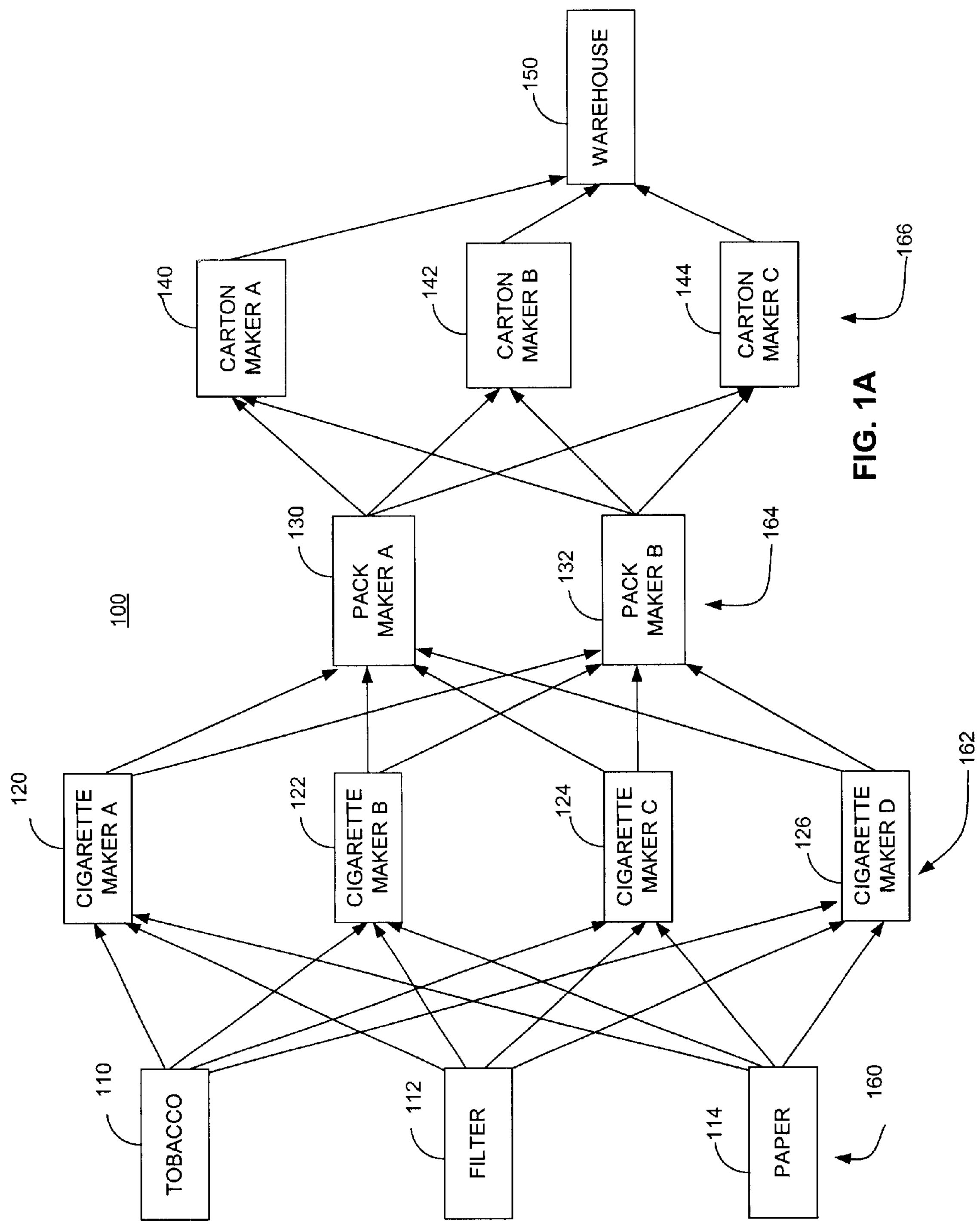

100
WAREHOUSE
150
CARTON MAKER A
140
CARTON MAKER B
142
CARTON MAKER C
144
166
PACK MAKER A
130
PACK MAKER B
132
164
CIGARETTE MAKER A
120
CIGARETTE MAKER B
122
CIGARETTE MAKER C
124
CIGARETTE MAKER D
126
162
TOBACCO
110
FILTER
112
PAPER
114
160

FIG. 1A

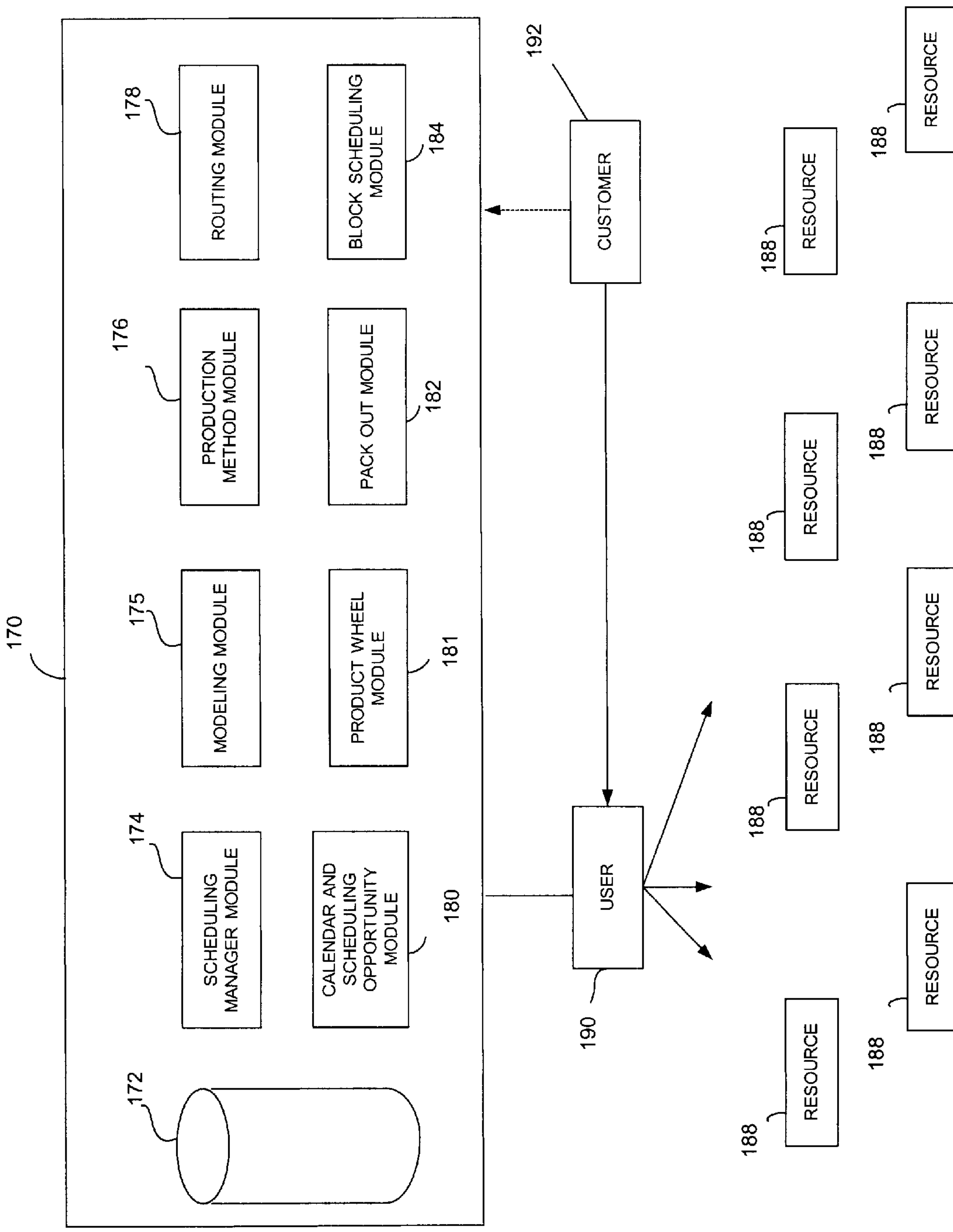
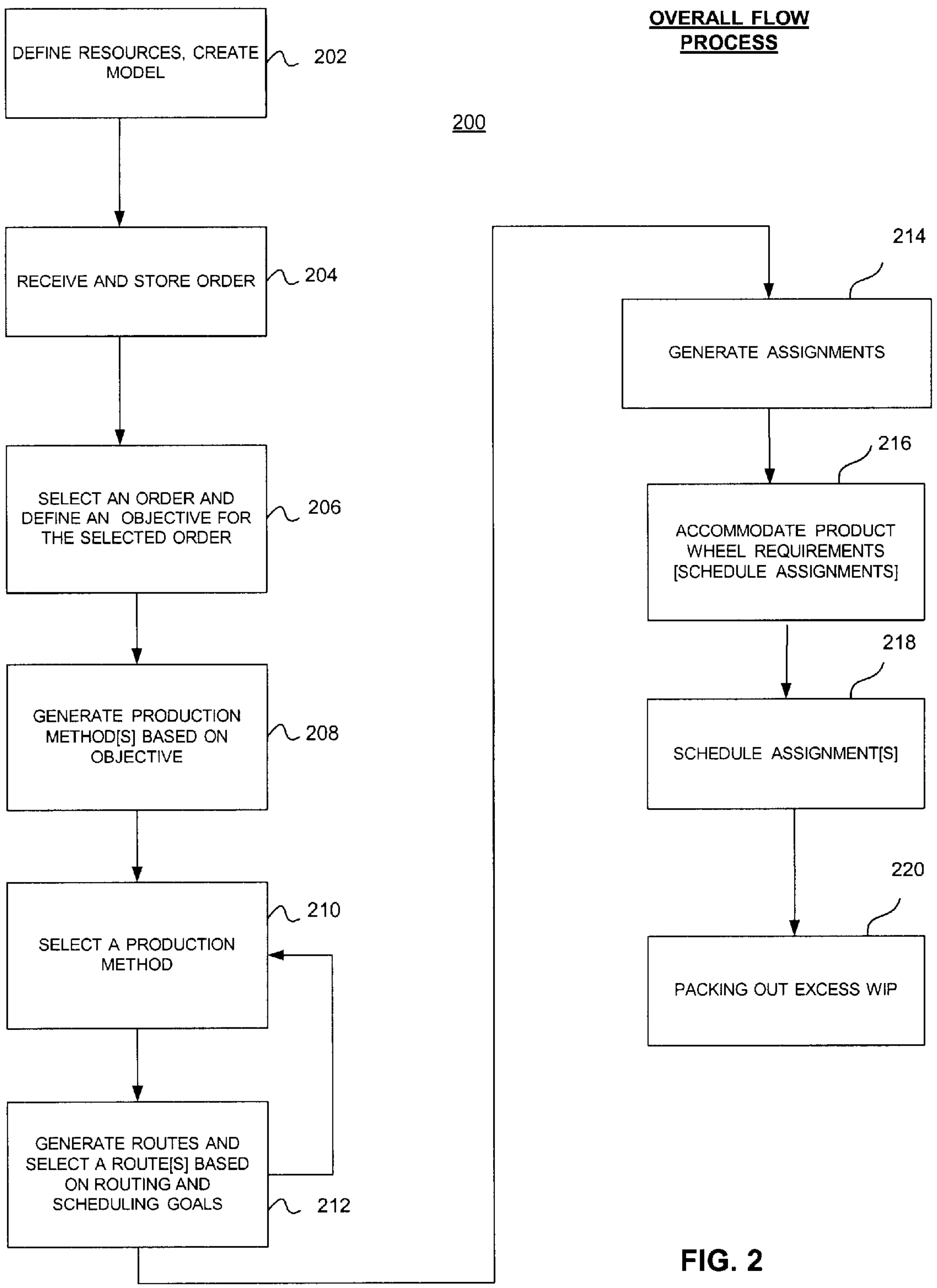

170
172
174
SCHEDULING MANAGER MODULE
175
MODELING MODULE
176
PRODUCTION METHOD MODULE
178
ROUTING MODULE
180
CALENDAR AND SCHEDULING OPPORTUNITY MODULE
181
PRODUCT WHEEL MODULE
182
PACK OUT MODULE
184
BLOCK SCHEDULING MODULE
192
CUSTOMER
190
USER
188
RESOURCE
188
RESOURCE
188
RESOURCE
188
RESOURCE
188
RESOURCE
188
RESOURCE
188
RESOURCE
188
RESOURCE

FIG. 1B



OVERALL FLOW PROCESS
200
DEFINE RESOURCES, CREATE MODEL
202
RECEIVE AND STORE ORDER
204
SELECT AN ORDER AND DEFINE AN OBJECTIVE FOR THE SELECTED ORDER
206
GENERATE PRODUCTION METHOD[S] BASED ON OBJECTIVE
208
SELECT A PRODUCTION METHOD
210
GENERATE ROUTES AND SELECT A ROUTE[S] BASED ON ROUTING AND SCHEDULING GOALS
212
GENERATE ASSIGNMENTS
214
ACCOMMODATE PRODUCT WHEEL REQUIREMENTS [SCHEDULE ASSIGNMENTS]
216
SCHEDULE ASSIGNMENT[S]
218
PACKING OUT EXCESS WIP
220

FIG. 2

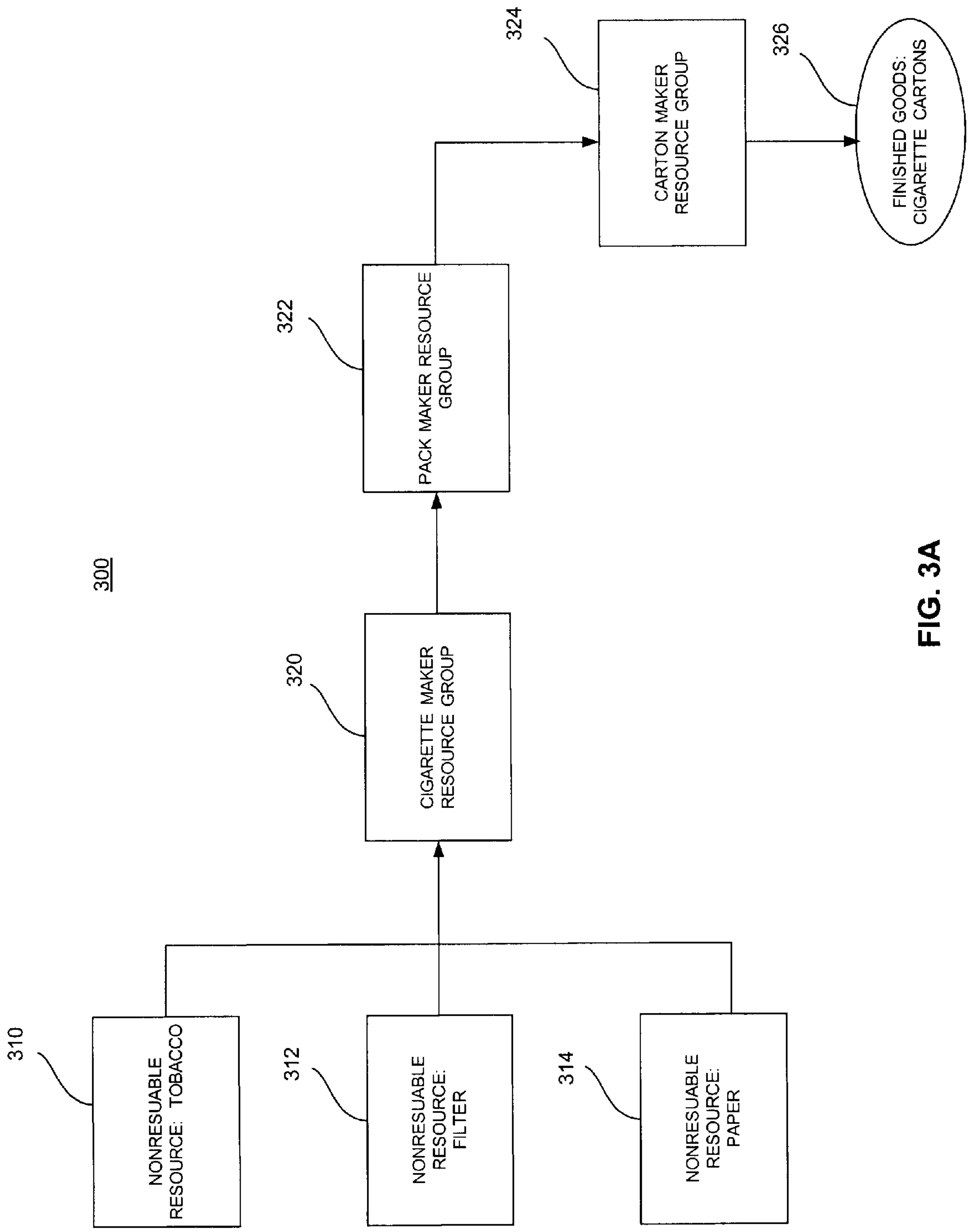

300
310
NONRESUABLE RESOURCE: TOBACCO
312
NONRESUABLE RESOURCE: FILTER
314
NONRESUABLE RESOURCE: PAPER
320
CIGARETTE MAKER RESOURCE GROUP
322
PACK MAKER RESOURCE GROUP
324
CARTON MAKER RESOURCE GROUP
326
FINISHED GOODS: CIGARETTE CARTONS

FIG. 3A

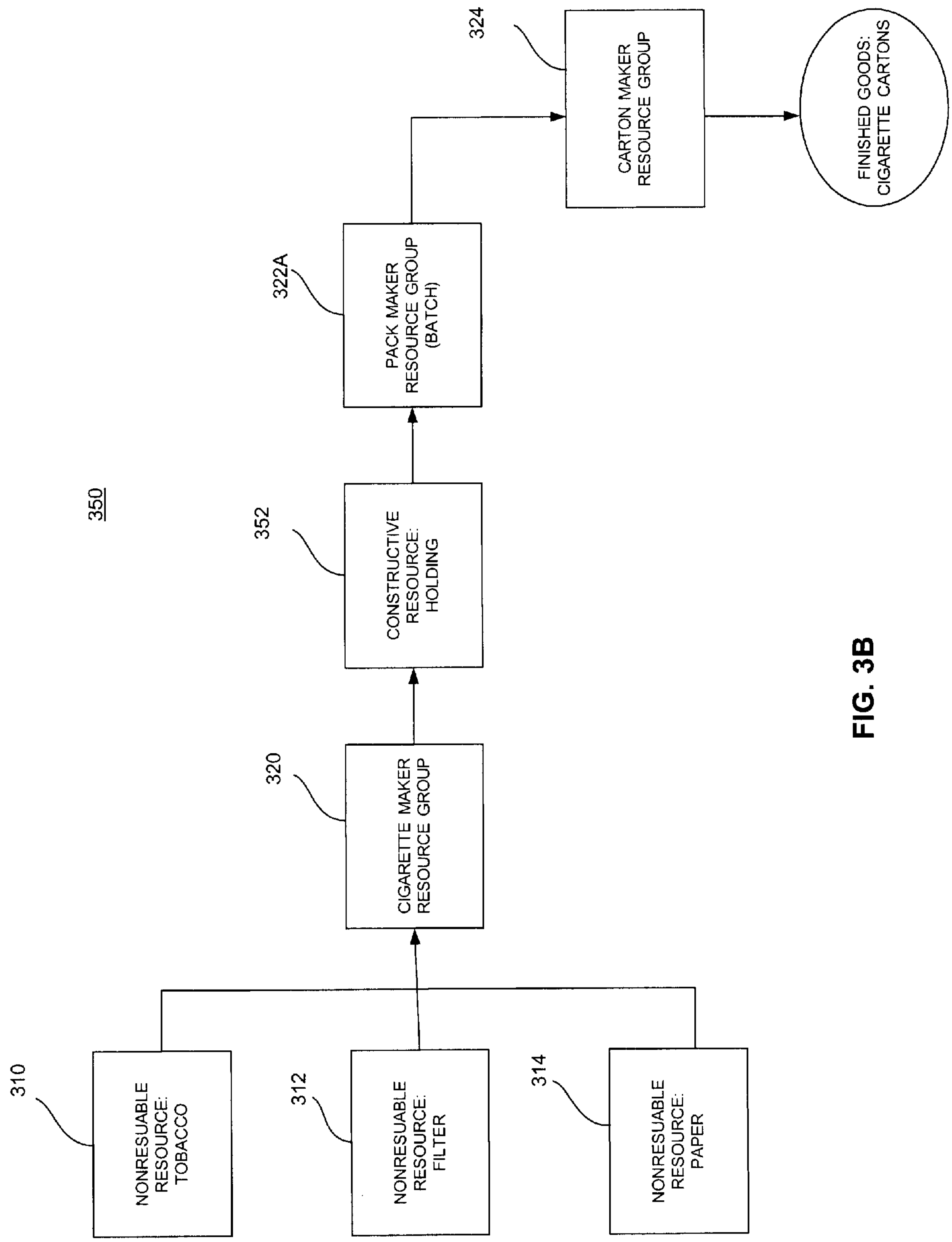

350
310
NONRESUABLE RESOURCE: TOBACCO
312
NONRESUABLE RESOURCE: FILTER
314
NONRESUABLE RESOURCE: PAPER
320
CIGARETTE MAKER RESOURCE GROUP
352
CONSTRUCTIVE RESOURCE: HOLDING
322A
PACK MAKER RESOURCE GROUP (BATCH)
324
CARTON MAKER RESOURCE GROUP
FINISHED GOODS: CIGARETTE CARTONS

FIG. 3B

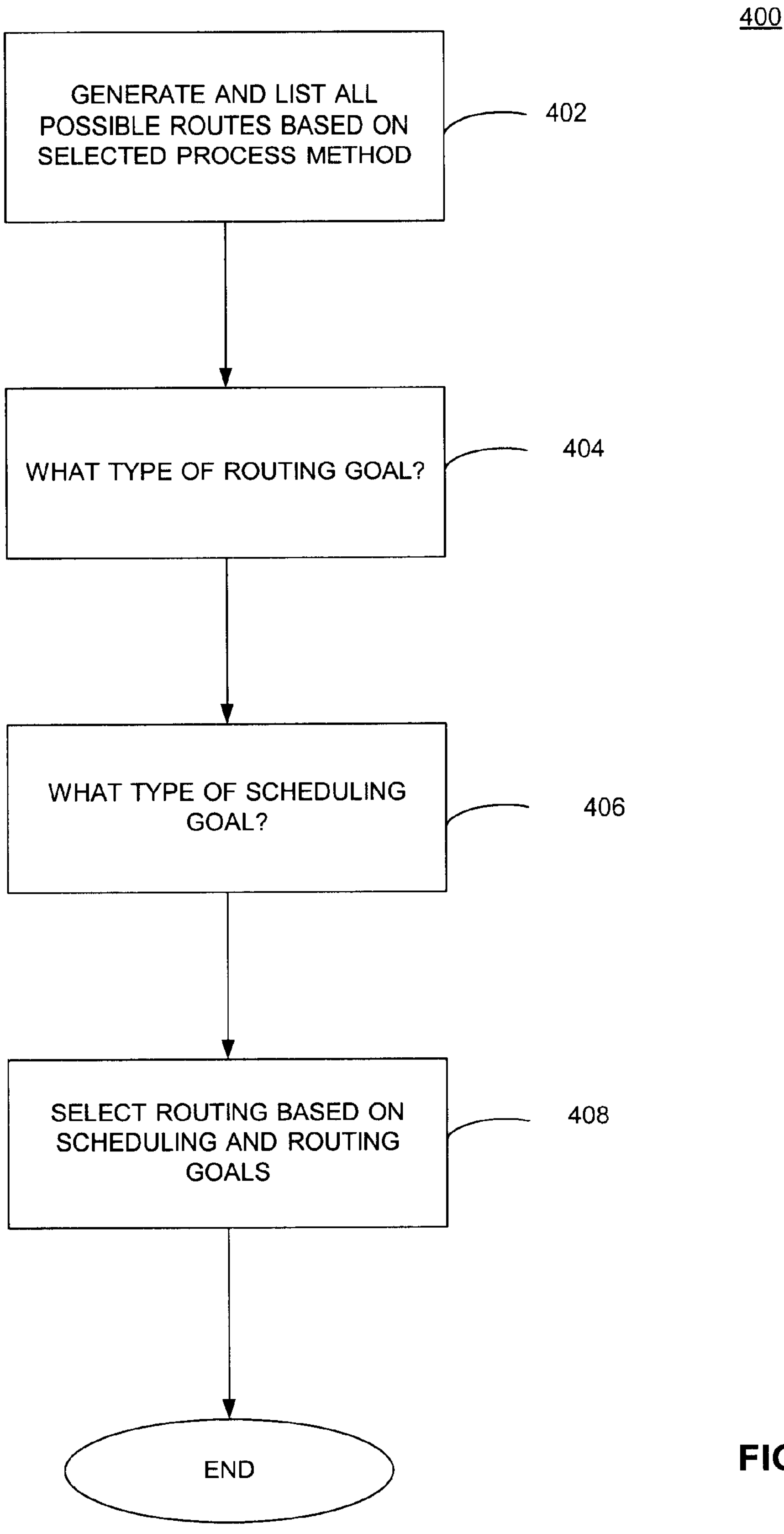
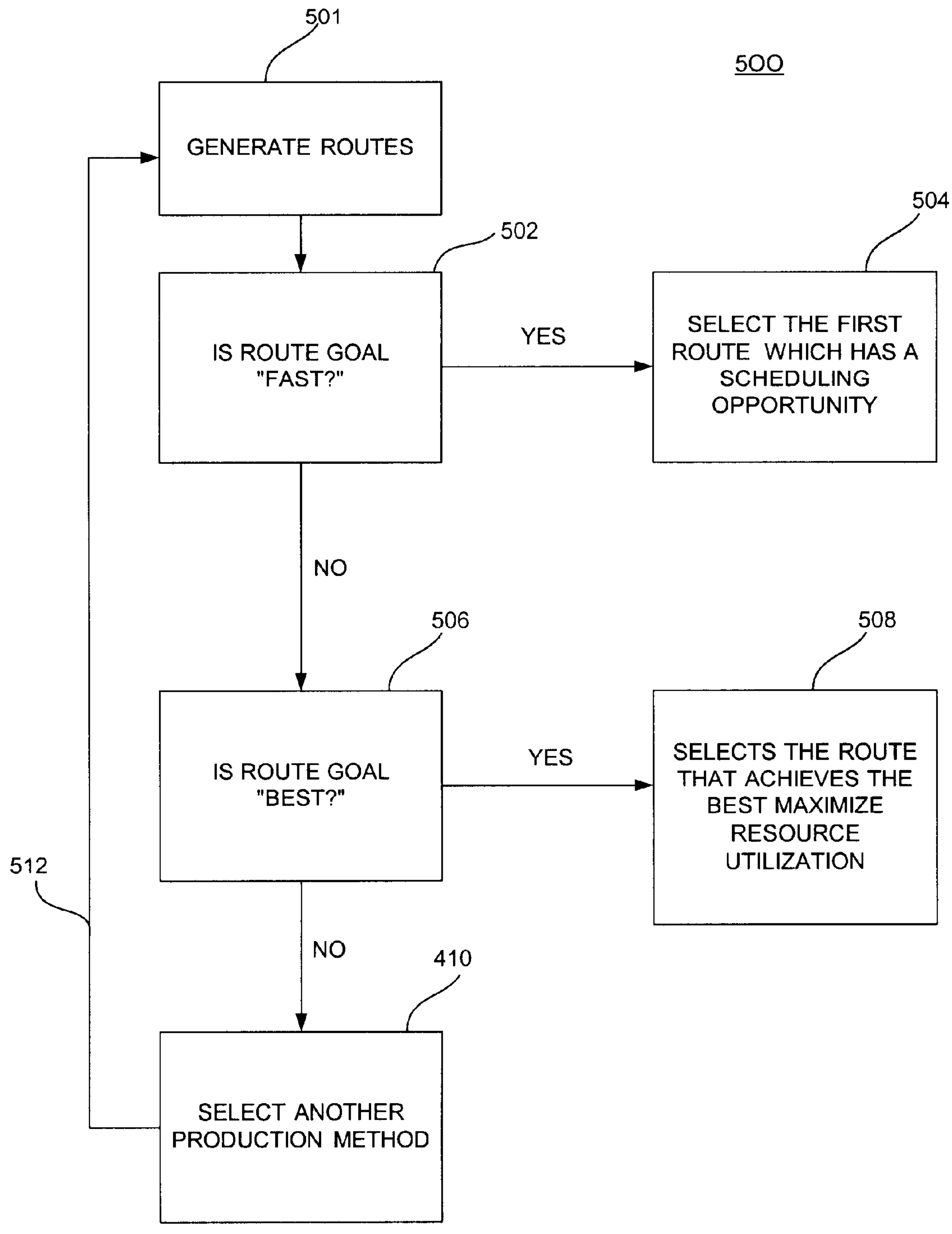

400
GENERATE AND LIST ALL POSSIBLE ROUTES BASED ON SELECTED PROCESS METHOD
402
WHAT TYPE OF ROUTING GOAL?
404
WHAT TYPE OF SCHEDULING GOAL?
406
SELECT ROUTING BASED ON SCHEDULING AND ROUTING GOALS
408
END

FIG. 4



500
501
GENERATE ROUTES
502
IS ROUTE GOAL "FAST?"
YES
504
SELECT THE FIRST ROUTE WHICH HAS A SCHEDULING OPPORTUNITY
NO
506
IS ROUTE GOAL "BEST?"
YES
508
SELECTS THE ROUTE THAT ACHIEVES THE BEST MAXIMIZE RESOURCE UTILIZATION
512
NO
410
SELECT ANOTHER PRODUCTION METHOD

FIG. 5

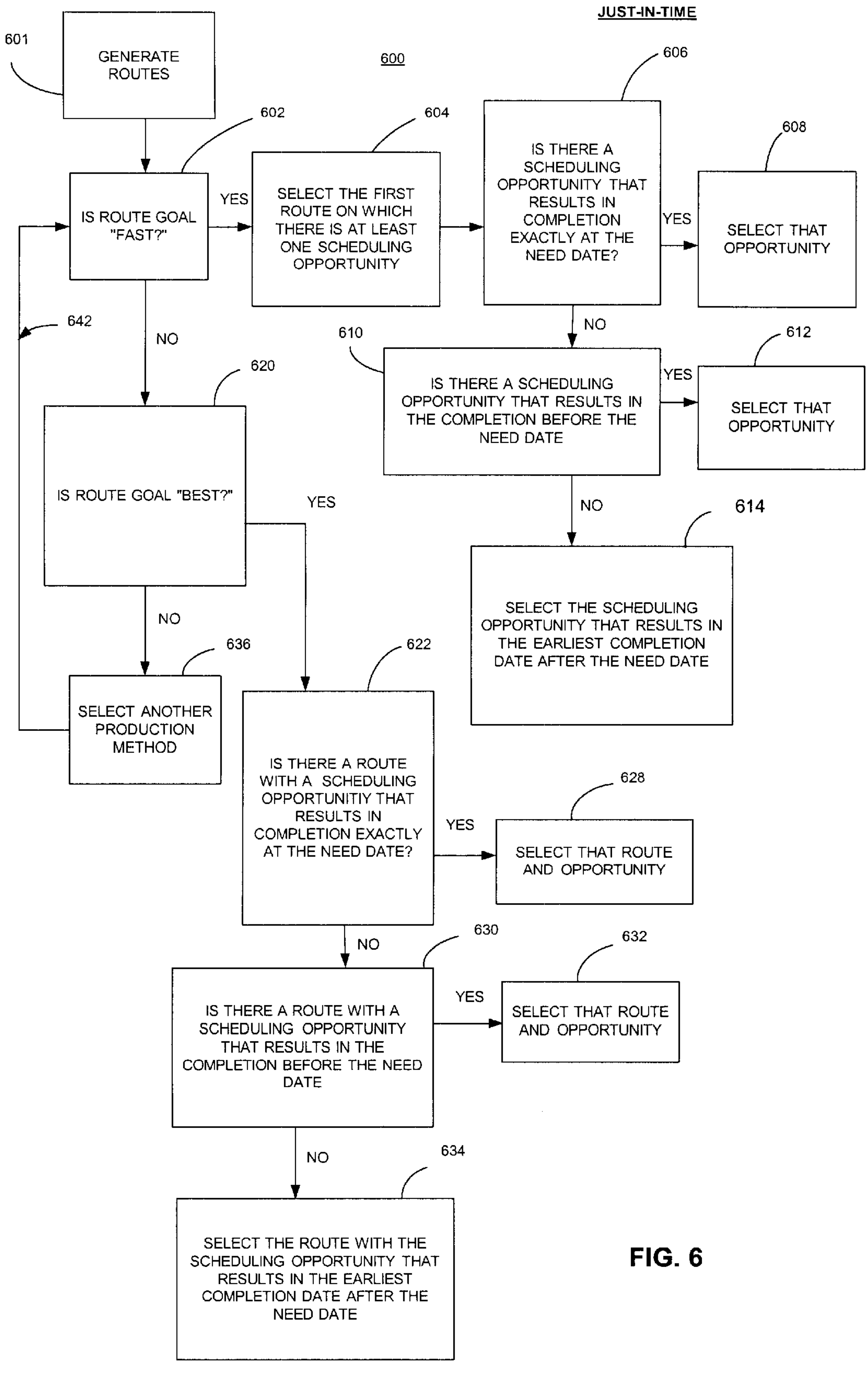

JUST-IN-TIME
600
601
GENERATE ROUTES
602
IS ROUTE GOAL "FAST?"
YES
604
SELECT THE FIRST ROUTE ON WHICH THERE IS AT LEAST ONE SCHEDULING OPPORTUNITY
606
IS THERE A SCHEDULING OPPORTUNITY THAT RESULTS IN COMPLETION EXACTLY AT THE NEED DATE?
YES
608
SELECT THAT OPPORTUNITY
NO
610
IS THERE A SCHEDULING OPPORTUNITY THAT RESULTS IN THE COMPLETION BEFORE THE NEED DATE
YES
612
SELECT THAT OPPORTUNITY
NO
614
SELECT THE SCHEDULING OPPORTUNITY THAT RESULTS IN THE EARLIEST COMPLETION DATE AFTER THE NEED DATE
642
NO
620
IS ROUTE GOAL "BEST?"
YES
NO
636
SELECT ANOTHER PRODUCTION METHOD
622
IS THERE A ROUTE WITH A SCHEDULING OPPORTUNITIY THAT RESULTS IN COMPLETION EXACTLY AT THE NEED DATE?
YES
628
SELECT THAT ROUTE AND OPPORTUNITY
NO
630
IS THERE A ROUTE WITH A SCHEDULING OPPORTUNITY THAT RESULTS IN THE COMPLETION BEFORE THE NEED DATE
YES
632
SELECT THAT ROUTE AND OPPORTUNITY
NO
634
SELECT THE ROUTE WITH THE SCHEDULING OPPORTUNITY THAT RESULTS IN THE EARLIEST COMPLETION DATE AFTER THE NEED DATE

FIG. 6

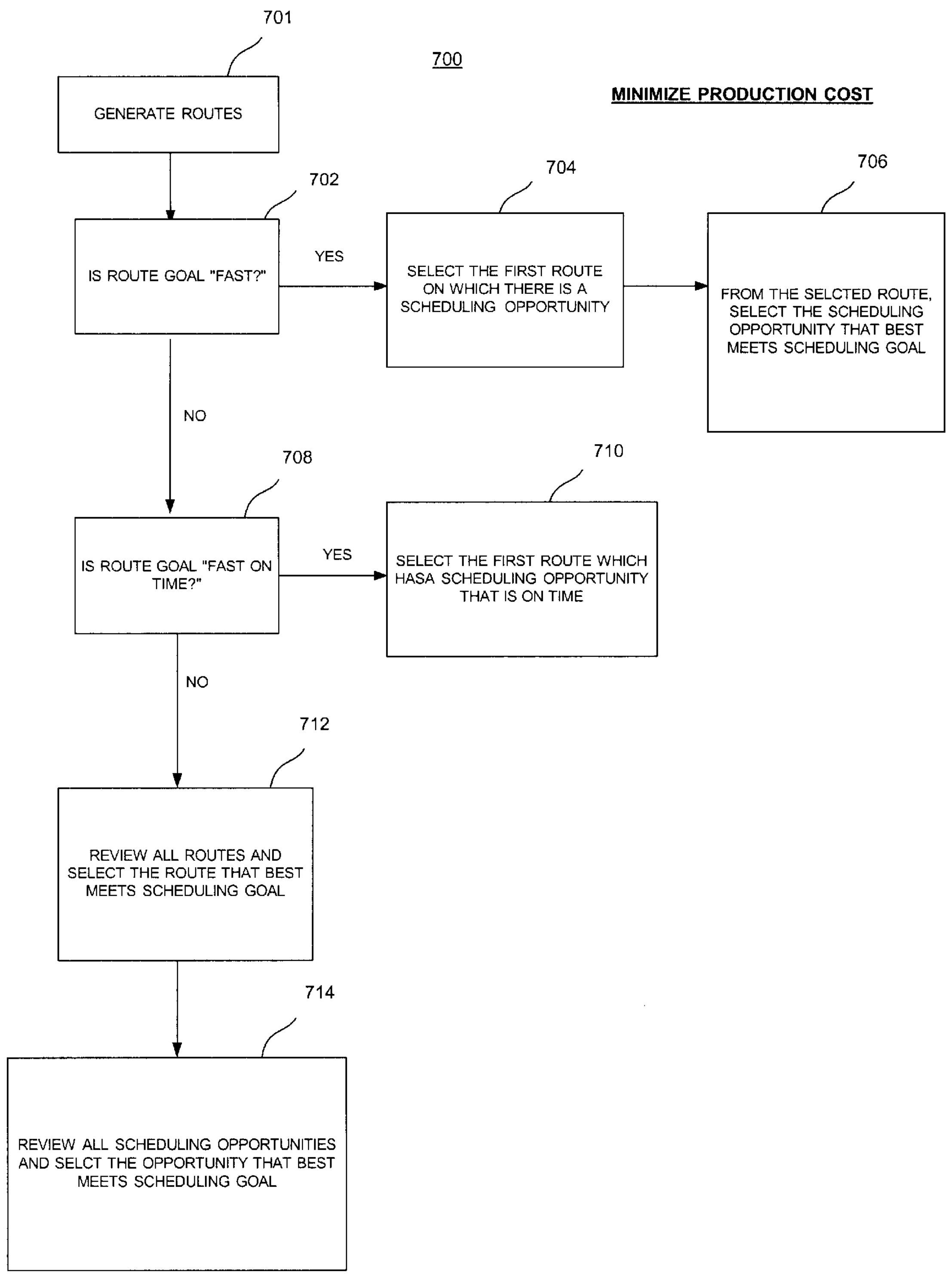

701
700
MINIMIZE PRODUCTION COST
GENERATE ROUTES
702
704
706
IS ROUTE GOAL "FAST?"
YES
SELECT THE FIRST ROUTE ON WHICH THERE IS A SCHEDULING OPPORTUNITY
FROM THE SELCTED ROUTE, SELECT THE SCHEDULING OPPORTUNITY THAT BEST MEETS SCHEDULING GOAL
NO
708
710
IS ROUTE GOAL "FAST ON TIME?"
YES
SELECT THE FIRST ROUTE WHICH HASA SCHEDULING OPPORTUNITY THAT IS ON TIME
NO
712
REVIEW ALL ROUTES AND SELECT THE ROUTE THAT BEST MEETS SCHEDULING GOAL
714
REVIEW ALL SCHEDULING OPPORTUNITIES AND SELCT THE OPPORTUNITY THAT BEST MEETS SCHEDULING GOAL

FIG. 7

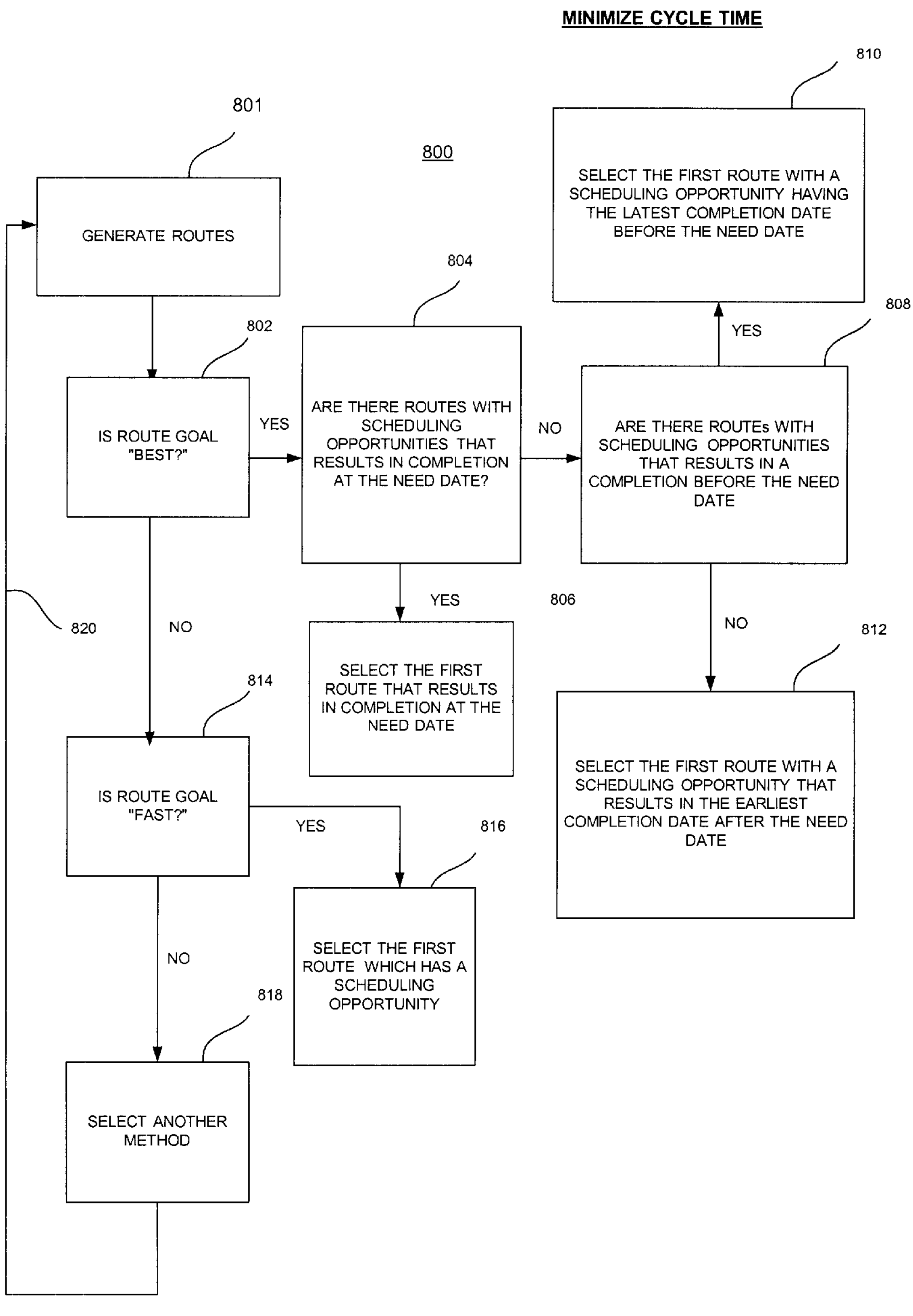

MINIMIZE CYCLE TIME
800
801
GENERATE ROUTES
802
IS ROUTE GOAL "BEST?"
YES
804
ARE THERE ROUTES WITH SCHEDULING OPPORTUNITIES THAT RESULTS IN COMPLETION AT THE NEED DATE?
NO
808
ARE THERE ROUTEs WITH SCHEDULING OPPORTUNITIES THAT RESULTS IN A COMPLETION BEFORE THE NEED DATE
YES
810
SELECT THE FIRST ROUTE WITH A SCHEDULING OPPORTUNITY HAVING THE LATEST COMPLETION DATE BEFORE THE NEED DATE
YES
806
SELECT THE FIRST ROUTE THAT RESULTS IN COMPLETION AT THE NEED DATE
NO
812
SELECT THE FIRST ROUTE WITH A SCHEDULING OPPORTUNITY THAT RESULTS IN THE EARLIEST COMPLETION DATE AFTER THE NEED DATE
820
NO
814
IS ROUTE GOAL "FAST?"
YES
816
SELECT THE FIRST ROUTE WHICH HAS A SCHEDULING OPPORTUNITY
NO
818
SELECT ANOTHER METHOD

FIG. 8

900

BALANCE RESOURCE UTILIZATION

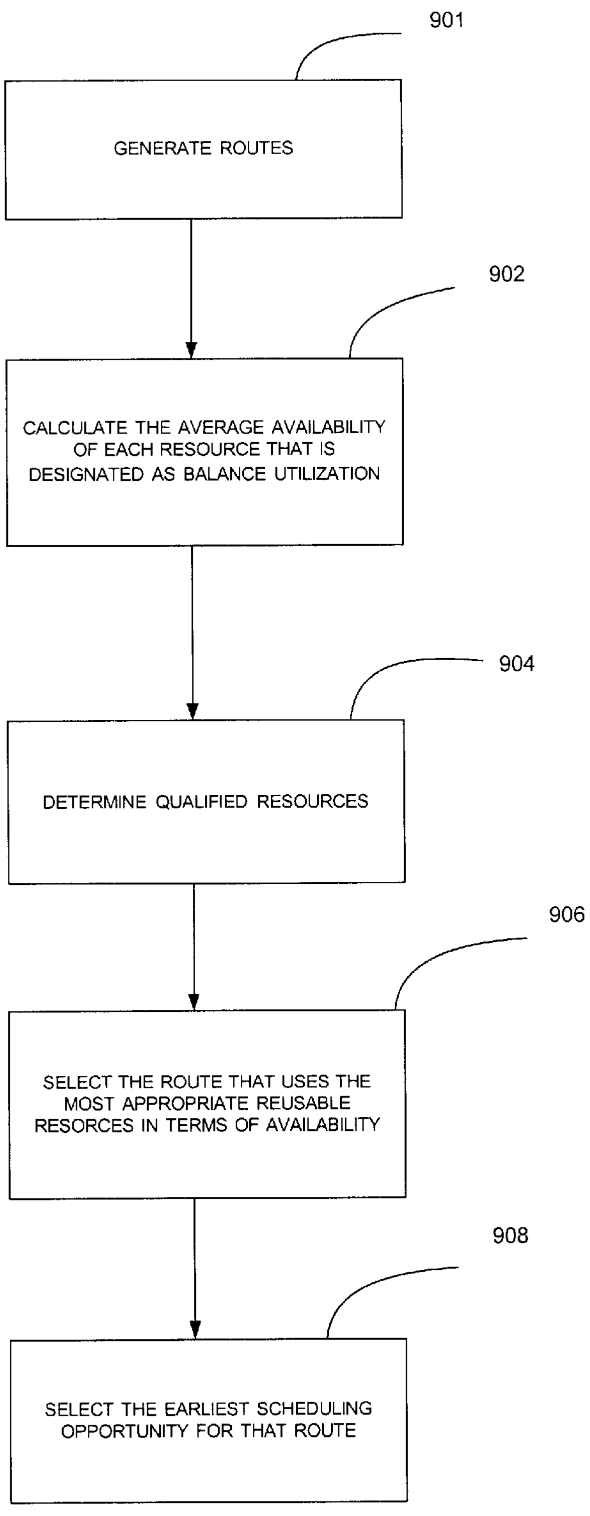
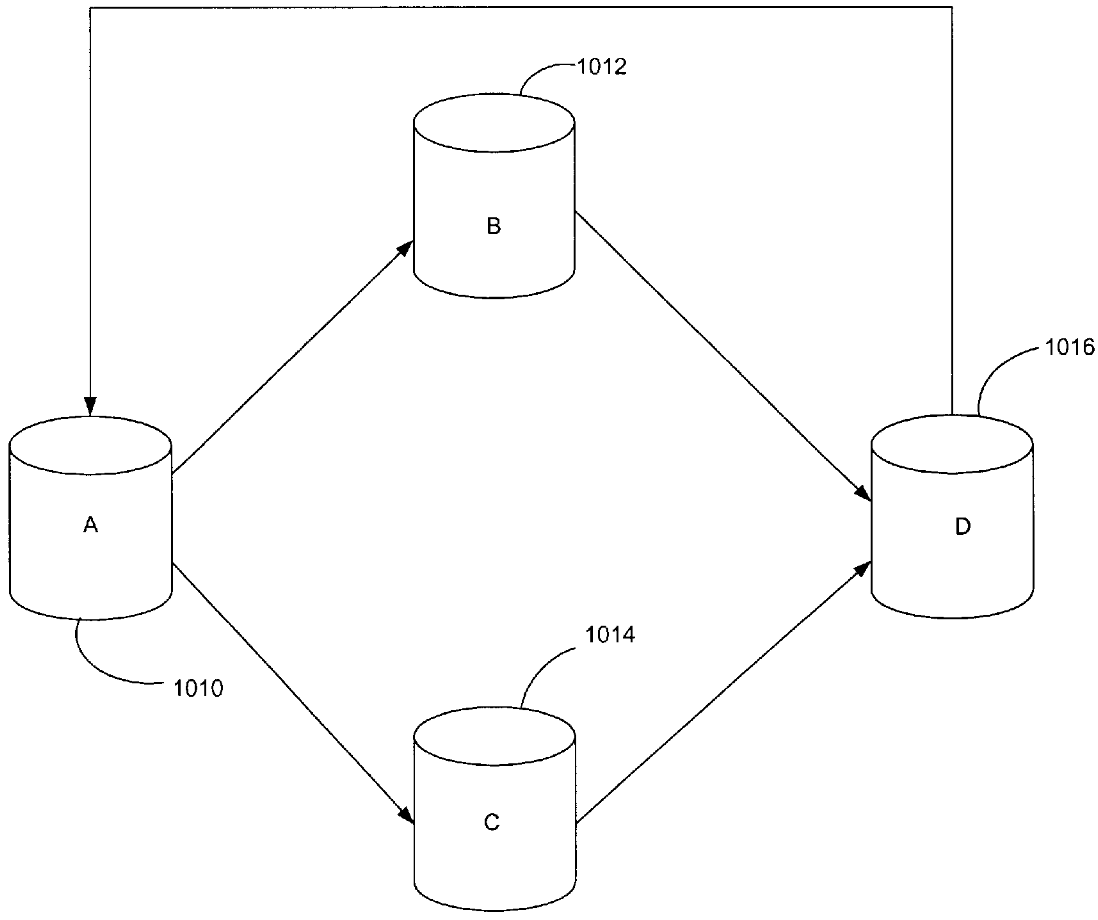


FIG. 9



Product Wheel Example

FIG. 10A

PRODUCT WHEEL EXAMPLE PART II

1020

PRODUCT WHEEL NAME	FROM SKU	TO SKU	COST	
OIL MIXER 1	A	B	20.00	1030
OIL MIXER 1	A	C	30.00	1032
OIL MIXER 1	B	D	3.00	1034
OIL MIXER 1	C	D		1036
OIL MIXER 1	D	A		1038
PACKAGER 1	A	B		1040
1050	1052	1054	1056	

FIG. 10B

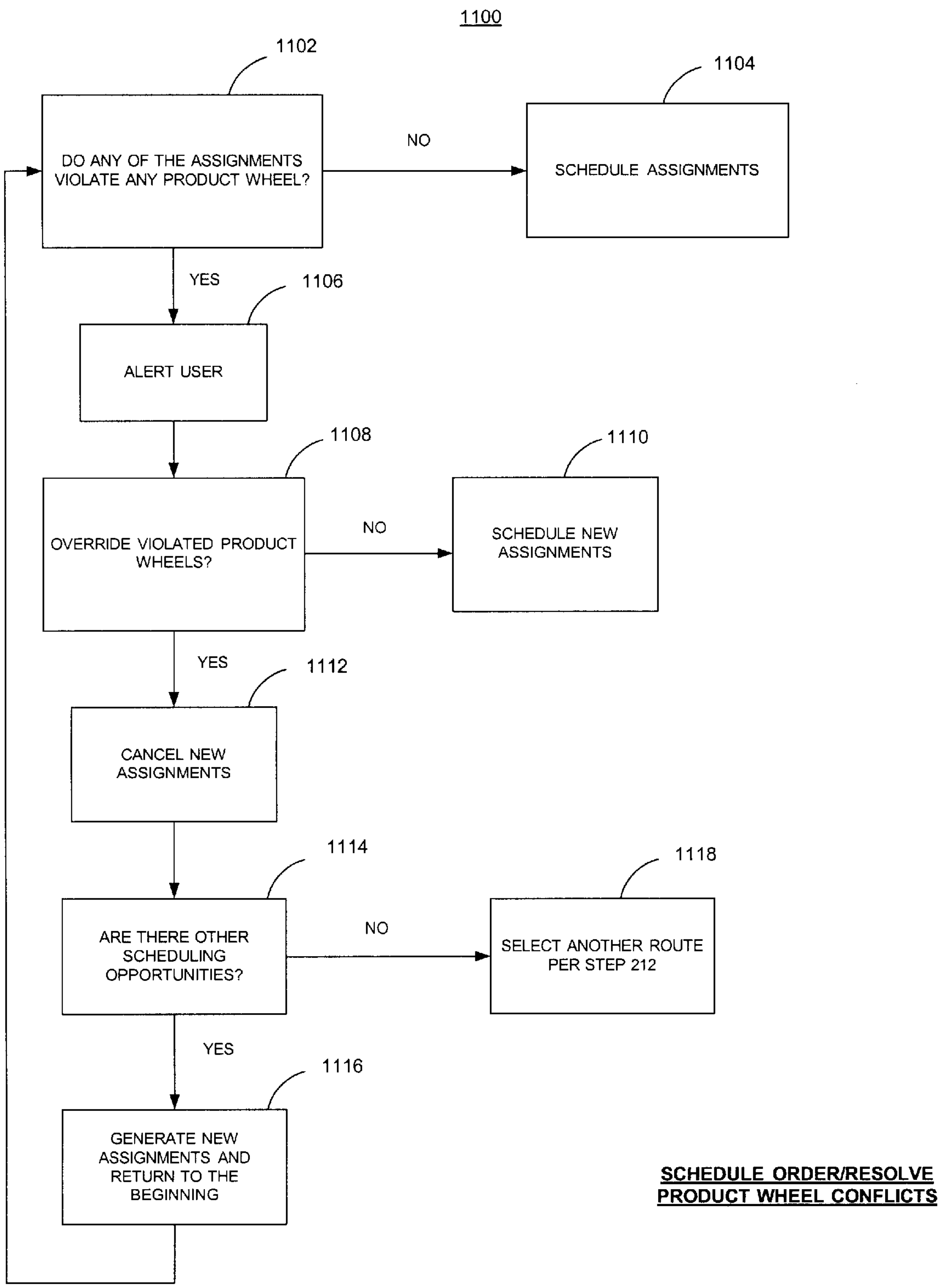

1100
1102
DO ANY OF THE ASSIGNMENTS VIOLATE ANY PRODUCT WHEEL?
NO
1104
SCHEDULE ASSIGNMENTS
YES
1106
ALERT USER
1108
OVERRIDE VIOLATED PRODUCT WHEELS?
NO
1110
SCHEDULE NEW ASSIGNMENTS
YES
1112
CANCEL NEW ASSIGNMENTS
1114
ARE THERE OTHER SCHEDULING OPPORTUNITIES?
NO
1118
SELECT ANOTHER ROUTE PER STEP 212
YES
1116
GENERATE NEW ASSIGNMENTS AND RETURN TO THE BEGINNING
SCHEDULE ORDER/RESOLVE PRODUCT WHEEL CONFLICTS

FIG. 11

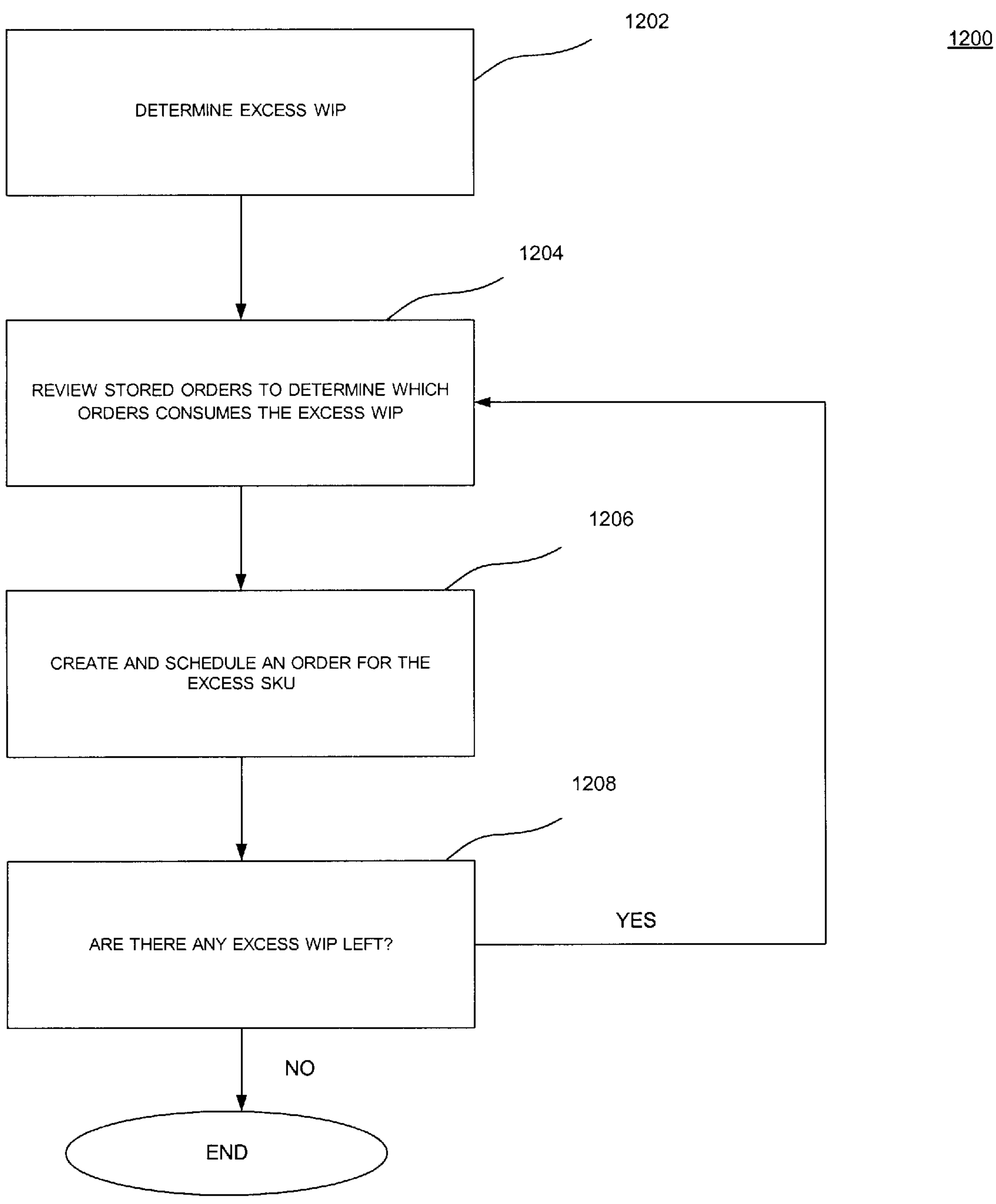

1200
1202
DETERMINE EXCESS WIP
1204
REVIEW STORED ORDERS TO DETERMINE WHICH ORDERS CONSUMES THE EXCESS WIP
1206
CREATE AND SCHEDULE AN ORDER FOR THE EXCESS SKU
1208
ARE THERE ANY EXCESS WIP LEFT?
YES
NO
END

FIG. 12

SYSTEM AND METHODS FOR SCHEDULING MANUFACTURING RESOURCES

CROSS REFERENCE TO RELATED APPLICATION

This application claims priority from U.S. Provisional Patent Application Serial No. 60/239,280, filed Oct. 12, 2000, the disclosure of which is hereby incorporated by reference by its entirety.

FIELD OF THE INVENTION

The invention relates to a system and method for solving manufacturing-scheduling (MS) problems. More specifically, the invention relates to a robust system and method for scheduling and planning manufacturing facilities and equipment based on pre-defined rules and goals.

BACKGROUND OF THE INVENTION

Scheduling of manufacturing resources, for example, materials, machinery, man hours, and the like, in response to customer orders has been a significant concern and problem in the field of manufacturing for almost as long as products have been manufactured. The best use of manufacturing facilities and equipment results in greater productivity and profit. The vast majority of MS problems have been resolved by humans without the aid of computer tools, particularly in businesses where the maximum use of the manufacturing equipment and facilities has not been essential to the success of the business.

The increasing size of businesses in general, the increased competition and the need to provide improved customer service and customization have magnified the difficulties of MS problems to the point where efficient human resolution of the issues posed has become impossible or at best highly inefficient. Consequently, MS software tools and process have been developed to address these issues.

Existing MS software tools and processes can be extremely complex to use and understand, and, in general, do not adapt conveniently to a variety of different businesses in an efficient way to address all of the specific issues particular to each individual business. Further, many of these existing tools lack the capability of recognizing and adapting to the specific needs of individual businesses. This is, in part, because of the complex nature of trying to solve MS problems. From a generic standpoint, the complexity of the MS problem results in the interaction of a relatively large number of factors necessary to create an adequate solution to the MS problem. The complexity of the MS software development process to accommodate the magnitude of issues involved, and the complexity of MS modeling of the business which is emulated by the software, have not permitted previous MS software processes to be as successful as expected. As a consequence, many prior MS software application have required the business characteristics to fit a relatively fixed model, rather than allow the individual business characteristics to form the model. The complexity of an MS program is considerable, typically requiring tens or hundreds of thousands of lines of software code. Writing an extensive amount of software code creates opportunities for numerous errors, thereby requiring extensive trial and error use to eliminate the unforeseen errors, which may only be revealed from such use. Therefore, writing additional code is not a desirable approach to solving individual MS problems.

This complexity exists despite the fact that many of the basic concepts involved in the MS model itself seem almost intuitive. Concepts, which may appear intuitive in MS situations, are usually accomplished in MS software only with considerable difficulty, effort and creativity.

For example, many businesses have multiple concurrent goals at any given time. The goals of a business will often be conflicting. To illustrate, many companies proscribe to or at least attempt to proscribe to the concept of just-in-time. Businesses that follow the just-in-time concept typically manufacture and deliver products just-in-time to meet the customer's due date. This allows manufacturers to minimize inventory reducing manufacturing and storage costs. However, many of these manufacturers may also like to maximize the use of their equipment or resources. That is, manufacturers prefer to minimize the idle time of their equipment or resources. Unfortunately, trying to merge these two goals will typically make the MS problem even more complex. In fact, sometimes these two goals may actually conflict with each other. Attempting to resolve conflicting goals of a business often produces mixed results.

Creating functional MS models has always been somewhat difficult and tedious. An MS model typically attempts to define and map the various resources available in a manufacturing facility. Unfortunately models for previous MS processes did not accurately represent the unique characteristic of resources. For example, the original MS processes were referred to as materials requirement planning (MRP). MRP is generally regarded as inadequate to meet current problems, primarily because MRP focused solely on the procedures for manufacturing a product and the timing associated with completion of the product.

One of the problems of MRP was that it was not very good at taking into account the limitations of resources. For example, MRP assumed an infinite capacity for each of the resources (e.g., work stations, tolls or people) available for use in the machining, assembly and production of a product. This assumption is simply not realistic. In reality, real resources have finite restrictions on capacity, operational capability, operational environment, etc. Thus, MRP generally ignored the operational constraints that limited how one could execute the manufacturing process.

Many manufacturing goals are difficult to accommodate in MS processes. For example, manufacturers sometimes strongly prefer that certain groups of products be produced by the same resource or resources in a particular sequential order. This is because there may be certain cost, operational or time advantages in doing so. The sequencing of production activities for different products in such a manner is generally referred to as "product wheels."

As described above, many of the MS systems currently available are at best ineffective because of the often complex nature of manufacturing facilities, the often conflicting goals concurrently sought by manufacturers, and the constraints typically associated with manufacturing resources. Further, because of the extremely tight timeline that many manufacturers face daily, a robust MS system and method capable of scheduling manufacturing processes in real time would be highly desirable. Further, such a system and method would be even more valuable if it can take into account, the conflicting manufacturing goals, the complex relations between resources, and the resource constraints typically associated with a manufacturing facility.

SUMMARY OF THE INVENTION

To resolve the problems cited above, the present invention provides, among other things, a system and methods for

scheduling manufacturing resources. In general, the present invention provides for a system and methods that schedule manufacturing resources by defining available resources, generating production methods and routes, and selecting routes based on user-defined goals. Further, the present system may use production wheels, packing out and block scheduling methods to optimally schedule manufacturing resources.

In a preferred embodiment, manufacturing data such as customer and internal orders, resource data, production methods, resource calendars, user goals, product wheels, SKU data, and the like, is stored in a database or a plurality of databases. After receiving and storing orders, one of the orders is selected for scheduling. Each order has an objective associated with it that identifies, for example, the finish goods or work-in-process that is the goal of the order. An MS model is created and stored in the database by defining available resources. Resources available for use in manufacturing are defined by identifying the constraints associated with each of the resources. A production method is generated based on the objective associated with an order selected for scheduling and the constraints as defined by the MS model. The production method identifies the production steps needed to attain the objective. Based on the production method generated, one or more routes are generated. One of the generated routes is then selected for scheduling based on user defined scheduling and/or routing goals. At least five types of scheduling goals are possible: maximizing resource utilization, just-in-time, minimizing production cost, minimizing cycle time and balance resource utilization. At least three types of routing goals are possible: fast, best, and fast on time.

According to another embodiment, product wheels are created and employed to prevent undesirable transitions from occurring. A transition is the process that a resource or a plurality of resources must undergo between succeeding assignments. By employing product wheels, undesirable sequencing of assignments may be detected and/or avoided.

According to another embodiment of the present invention, excess work-in-process may be packed out. As a by-product of manufacturing processes, excess work-in-process may be generated. By monitoring for and identifying excess work-in-process, and identifying and scheduling stored orders that consumes the excess work-in-process, the excess work-in-process may be eliminated.

According to another embodiment, block scheduling may be employed to reserve one or more resources for a product family. Reserving a resource may be accomplished by creating block calendars that are associated with the resource. The block calendars are divided into time intervals and product families assigned to those periods to reserve the resource for the product families.

As will be readily appreciated by one of ordinary skill in the art, the present invention provides for a robust system and method for scheduling manufacturing resources. Additional features and advantages are set forth in the description that follows, and in part are apparent from the description, or may be learned by practice of the invention. The objectives and other advantages of the invention are realized and attained by the structure particularly pointed out in the written description and claims hereof as well as the appended drawings.

It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory and are intended to provide further explanation of the invention as claimed.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are included to provide further understanding of the invention and are incorporated in and constitute a part of this specification, illustrate embodiments of the invention and together with the description serve to explain the principles of the invention. In the drawings:

FIG. **1A** is an exemplary manufacturing facility for manufacturing cigarette cartons;

FIG. **1B** is a block diagram depicting one embodiment of the system for scheduling manufacturing activities;

FIG. **2** is a flow diagram depicting the general steps for scheduling manufacturing resources;

FIG. **3A** is a flow diagram of an exemplary production method for manufacturing cartons of filtered cigarettes;

FIG. **3B** is a flow diagram of a second exemplary production method for manufacturing cartons of filtered cigarettes;

FIG. **4** is a flow diagram for reviewing and selecting a feasible route or routes;

FIG. **5** is a flow diagram for selecting a route when the scheduling goal is maximizing resource utilization;

FIG. **6** is a flow diagram for selecting a route when the scheduling goal is just-in-time;

FIG. **7** is a flow diagram for selecting a route when the scheduling goal is minimizing production costs;

FIG. **8** is a flow diagram for selecting a route when the scheduling goal is minimizing cycle time;

FIG. **9** is a flow diagram for selecting a route when the scheduling goal is balance resource utilization;

FIG. **10A** is a block diagram depicting exemplary transitions allowed for four grades of oil;

FIG. **10B** is a chart depicting an exemplary product wheel;

FIG. **11** is a flow diagram for resolving product wheel conflicts; and

FIG. **12** is a flow diagram for packing out excess work-in-process.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

To explain the novel features of this invention, the following example of an exemplary process for manufacturing cigarettes is provided. FIG. **1A** illustrates the various routes that may be used for making and packaging cigarettes in an exemplary manufacturing facility.

The production of manufactured goods, such as cigarettes, is often initiated by an order from a customer. Such an order typically contains certain relevant information, for example, the item or items being ordered, the quantity and quality of the item or items, delivery location and the “need date.” The need date is the date that the customer must or prefers to have the ordered item[s] delivered. Based on the information contained in the order, a manufacturer must carefully plan the manufacturing process that meets the requirements of that order.

The process for making and packaging cigarettes begins when tobacco **110**, filter **112** and paper **114** are sent to one or more cigarette makers **120**, **122**, **124** and **126**. The tobacco **110**, filter **112** and paper **114** are non-reusable resources while cigarette makers **120**, **122**, **124** and **126** are reusable resources. Here, the tobacco, filter and paper resources **110**, **112** and **114** are non-reusable resources

because unless these resources are regularly replenished, the resources will eventually be depleted. On the other hand, the cigarette makers **120**, **122**, **124** and **126** are manufacturing equipment that is not replaced very often. After getting the raw materials (i.e., tobacco, paper and filter), the cigarette makers **120**, **122**, **124** and **126** makes cigarettes by rolling the tobacco **110** inside the paper **114** and inserting the filter **112** to the end of the rolled up tobacco and paper to produce cigarettes. Of course, all of the cigarette makers **120**, **122**, **124** and **126** may not be available for use at any given time. That is, several of the cigarette makers may be in use for other customer orders. For example, suppose in addition to the order for filtered cigarettes described above, there was a second order that was needed to be scheduled during the same time period, for example, a customer order for unfiltered cigarettes. In this situation, some of the cigarette makers **120**, **122**, **124** and **126** may be assigned to the second order, thus, making some of the cigarette makers unavailable for use in manufacturing filtered cigarette for the first order.

After the cigarettes are manufactured, they are sent to one or both cigarette pack makers **130** and **132**. The cigarettes pack makers **130** and **132** place twenty individual cigarettes into a package creating a single cigarette pack. Once a cigarette pack is made, they are then sent to one or more carton makers **140**, **142** and **144**. The carton makers **140**, **142** and **144** place twelve packs of cigarettes into a carton, seal the carton and then ship the carton out to a warehouse.

The arrows between the various resources in FIG. **1** are individual legs of different possible routes for turning raw materials (i.e., tobacco, paper and filters) into cartons of cigarettes. Note that although the number of resources depicted in FIG. **1** is relatively few, many different routes are possible. Thus, the variation and complexity of routes exponentially increases with the number of resources available.

Each route is generally associated with a unique combination of resources. Along any given route, only specific resources are utilized. For example, in one route for producing cigarette cartons, the following resources are utilized, tobacco **110**, filter **112**, paper **114**, cigarette maker B **122**, pack maker B **132**, and carton maker C **144**. Meanwhile, another route may utilize the following resources, tobacco **110**, filter **112**, paper **114**, cigarette maker C **124**, pack maker A **130**, and carton maker A **140**.

In addition to all the different route variations, manufacturers must should take into consideration "scheduling opportunities" when scheduling manufacturing resources in accordance with a customer order. A scheduling opportunity is the time slot or time interval when a particular resource is available for use. For example, suppose the manufacturing facility in our previous example is very busy. If a new customer order arrives that needs to be immediately satisfied, then the manufacturer must make sure that there are scheduling opportunities or open time slots available for each resource being used to meet the order. Thus, when a manufacturer creates a production plan (i.e., routing plan) for meeting a customer order, it should, in addition to planning the appropriate route, make sure that there are scheduling opportunities for all the resources needed for the planned route.

The process for making cigarettes illustrated above has been greatly simplified for explanatory purposes. That is, in reality, as more actual operating conditions are accounted for, the manufacturing plans quickly become much more complex. For example, many reusable resources, such as cigarette makers **120**, **122**, **124** and **126** may only be able to produce products in batches. If a batch of products being produced by a reusable resource (e.g., cigarette makers) is greater than a subsequent resource on the same route (e.g., pack maker), then the manufacturer must somehow account for the difference between succeeding resources. Further, manufacturers should also take into account the difference in job time requirements for resources along the same route. For example, suppose cigarette maker B **122** and pack maker A **130** are assigned to work together on the same route for a particular customer order. However, suppose the time it takes to make a pack of cigarette by the pack maker A **130** is much less than the amount of cigarettes being produce by cigarette maker B **122**. In this situation, the manufacturer will have to decide whether to have the pack maker A **130** sit idle while waiting for cigarette maker B **122** to make enough cigarettes to make a pack or to run another customer order through the pack maker A **130** while the cigarette maker B **122** is making enough cigarettes to make a pack. As the above example illustrates, the planning of manufacturing products in a manufacturing facility is often complex, tedious and time consuming.

The resources depicted in FIG. **1A** may be grouped into resource families or groups **160**, **162**, **164** and **166**. Tobacco **110**, filter **112** and paper **114** make up a nonreusable resource family (or group) **160** for raw materials. The cigarette makers **120**, **122**, **124** and **126** make up a reusable resource family **162** for cigarette making machines. The pack makers **130** and **132** make up a second reusable resource family (or group) **164**, this one for cigarette pack making machines. Finally, the carton makers **140**, **142** and **144** make up a third reusable resource family (or group) **166**, this one for cigarette carton making machines. Note that each member of a family have common characteristics that are shared by all of the members of the family. Sometimes members of a family will perform the same type of function within the manufacturing plant. However, in other families, members may not perform the same functions. For example, for the nonreusable resource family for raw materials **160**, the tobacco resource **110** will not be able to typically replace the filter resource **112**.

Similarly, products or stock keeping units (herein "SKU") may also be grouped into product families. In our example, cigarettes may be divided into filtered or unfiltered cigarettes. The filtered cigarettes may then be sub-divided into subgroups, for example, menthol and non-menthol. Often, manufacturers prefer to manufacture products by manufacturing members of the same product family using the same resources and/or in a specific sequential order. Scheduling the production of a product family in such a manner is generally known as "campaigns" or "product wheels." Manufacturers often prefer using campaigns during production because there are certain advantages of producing products in such a way, for example, reduced manufacturing costs.

According to the present invention, there is provided a robust system and method for planning and scheduling manufacturing resources using a goal oriented scheduling process. FIG. **1B** illustrates a system **170** according to one embodiment of the present invention. The system **170** may be embodied in, for example, a work station, a server, a network of computer devices, or the like. The system may be located at a manufacturing facility or may be remotely located away from the facility. The system **170** comprises a database **172** and several modules **174** to **184** (described below). Although the database **172** in FIG. **1B** is depicted as being a single database **172**, it could of course be several databases embodied in multiple servers or workstations. The database **172** may be used to store various data including a

customer order, internally generated orders, models that define the resources available, production methods, resource attributes and constraints, resource calendars, user goals, product wheels, SKU data, and the like, all of which is described below. The data, stored in the database **172**, may be imported from external software applications, such as a supply planning application. For example, the Networks Demand application by Manugistics, Inc.™ may supply relevant data to the system **170** to be stored and parse during the employment of the various system features. The user **190**, who may be the manufacturer or any other entity or person responsible for managing the manufacturing facility, is electronically in communication with the system (for example by wireless or wire channels). A customer **192** may be in direct communication with the system **170** and/or in communication with the user **190**. Thus, the customer **192** may place orders directly with the system **170** or through the user **190**. Based on the customer order and user goals, the system **170** is able to schedule manufacturing resources **188** for production processes that fulfill the requirements of the customer order and user goals. The resources may be plant equipment, raw materials, work-in-process (herein "WIP"), man-hours, storage items, transportation items, or any other resources associated with a manufacturing facility.

The system **170**, when used for scheduling the resources, employs various modules **174** to **184** individually or in combination. Briefly, a scheduling manager module **174** receives, organizes and distributes the various data stored in the data database **172**. For example, the scheduling manager module **174** stores the orders received from customers **192** and may organize the orders in a particular order such as by need date or date received. Further, the scheduling manager module **174** identifies the scheduling and routing goals of the user **172** and stores/retrieves the goals in the database **172**. Based on the scheduling and routing goals of user **172**, the scheduling manager module **174** selects a route[s] for scheduling. Based on the selected route, the scheduling manager module **174** selects scheduling opportunities for each resource associated with the selected route. The scheduling manager module **174** may also monitor for excess SKUs (SKU may be finished goods or work-in-process) and schedule orders to consume the excess SKUs.

A modeling module **175** assists the user **190** in creating an MS model (herein "model") by defining the resources available for use. The resources may be defined by defining the various constraints associated with each of the resources. For example, attributes such as capacity, continuous or batch production capabilities, and any other attributes relevant to production processes may be used to define resources. Further, the modeling module **174** may also define SKUs, both finished goods and work-in-process, that may be present as a result of production activities. Attributes that may be assigned to the SKUs include, for example, whether the SKUs need to be packed out if excess SKUs exist and whether the SKUs are associated with any finished goods. A production method module **176** generates the production methods based on orders and resources available as identified by the model. Based on one of the production methods generated, and the user's scheduling and routing goals, a routing module **178** generates feasible route[s]. A calendar and scheduling opportunity module **180** creates and maintains calendars for each of the available resources defined in the model. A calendar, when used in association with a particular resource, is used to reserve or hold the resource for a particular assignment.

A product wheel module **181** creates and maintains product wheels. A product wheel is a set of one or more allowed product transitions associated with a given resource. The product wheel module **181** further prevents product wheels from being violated or at least alerts the user **190** that a wheel may be violated if a particular route is scheduled. This may be accomplished by reviewing routes and the scheduling opportunities selected by the scheduling manager **174** to determine whether they violate any product wheels. If a violation is detected, the scheduling manager **174** may cancel the selected route and scheduling opportunities or alert the user **190** that a product wheel is being violated by the selected route and scheduling opportunities.

A pack out module **182** monitors and detects any excess SKUs (for finished goods) or work-in-process (herein "WIP") that may exist and schedules orders that consumes the excess SKUs or WIPs. The block scheduling module **184** allows a user **190** to reserve resources to specific products or product families during specified time periods. The block scheduling module **184** can create block calendars and assign the calendars to a specific resource that the user wants dedicated to a particular product family. The functionality of the modules **174** to **184** described above may be best understood with the description of the following examples and processes.

Referring to FIG. **2** depicting a process, in accordance with one embodiment of the present invention, for scheduling manufacturing resources based on orders, in its most generic and high-level state. Orders may be customer or internally generated orders. Detailed description of the generic steps depicted in FIG. **2** are described in FIGS. **3** to **12**.

The process flow **200** in FIG. **2** involves generic steps which are applicable to essentially any type of business or manufacturing organization. For example, the process **200** may be applied to chemical manufacturing, consumer goods production, industrial equipment manufacturing, and the like. Users of the present invention may be, for example, various types of manufacturers, entities, or persons involved in the planning and/or scheduling of resources used to satisfy or fulfill customer or internal orders. The present invention is a scheduling system and methods that may be applied to many types of businesses in many situations.

Generally, the process **200** begins when a model is created by defining the resources available at step **202**. At step **204**, an order is received and stored. At step **206**, an order is selected and an objective defined based on the order. At step **208**, one or more production methods are generated based on the objective selected in the previous step. At step **210**, a production method is selected from the methods generated in step **208**. At step **212**, feasible routes are generated based on the method selected and a route is selected based on user goals. At step **214**, assignments are generated according to the route selected in step **212**. At step **216**, the process determines if any product wheels have been violated and if so, adjusts route and/or method accordingly (product wheels are described below). At step **218**, the assignments generated are scheduled. At step **220**, the process packs out any excess work-in-process ("WIP") that may be generated by the planned route. Although not shown in FIG. **2**, the system according to the present invention may also employ block scheduling. The steps described above will now be discussed in greater detail.

Preferably during the early stages of the process **200**, a model is created by identifying and defining both reusable and nonreusable resources that are available for use as depicted in step **202**. Resources that need to be identified include all of the resources available for use in any manu-

facturing activities contemplated by the system, for example, raw materials, individual reusable resources, and pooled reusable resources. Once the resources are identified, they must be defined. To define a resource, information related to the resource is provided such as the starting and ending SKUs being consumed or generated by the resource, the resource consumption and production rates, which family the resource belongs to, and any other constraints associated with that resource. Defining a resource also generally requires that the resource's relationship with other resources be defined. For instance, suppose pack maker B **132** is only able to physically accept the output from cigarette maker C **124** or cigarette maker D **126** because there is no assembly line connecting the pack maker B **132** to the other cigarette makers **120** and **122**. In such a situation, the pack maker B's limitations must clearly be defined in the model. Other resource constraints should also be identified, such as, whether a reusable resource's output is continuous or is incremental. That is, whether a reusable resource produces finished goods or WIPs continuously or whether it produces finished goods or WIPs in batches. For example, the cigarette makers **120**, **122**, **124** and **126** may only produce cigarettes in increments of forty cigarettes per batch.

Each resource is typically associated with a calendar. The calendar is used for scheduling individual activities related to specific orders. Activities are the series of specific process actions or steps required for producing the finished goods or WIPs sought by a specific order. The calendars may be divided into time slots. Each time slot may accommodate an order activity. The calendar may be customized so that it may be compatible with its associated resource. For example, the production capacity of pack makers **130** and **132** may be extremely high so that a calendar associated with one of the pack makers may be divided into small time slots, for example, 15-minute time intervals. This allows the pack makers **130** and **132** to be scheduled for many activities associated with many different orders during short time periods. On the other hand, calendars of resources having lower production capacity may be divided into longer time slots, for example, 60 minutes.

The process of scheduling an order generally begins when an order is received and stored in, for example, a database at step **204**. The order may originate from any source, for example, it may be a customer order or an order generated internally. The database may be remotely located in a standalone server or may be located in the same server where the system is located. The orders stored in the database may be organized in a number of different ways. For example, they may be stored according to the time and date that they were received or by clients or by the type of items being ordered.

According to one embodiment of the present invention, information related to the order information such as quantity and quality of goods and need date may all be stored. In addition, other information related to the order may be stored. For example, the type of WIPs that may be consumed during the manufacturing process called for by the order may also be defined and stored. This information may be used to execute the "packing out excess work-in-process" feature (described below) that may be incorporated into the system.

After orders are stored, the system selects an order and defines an objective for that order at step **206**. How the system selects an order among the plurality of orders that may be stored is accomplished by various means depending on user preferences. For example, if the orders stored are organized according to received date then the order having the earliest received date may be selected first. Once an order has been selected, the objective for that order is defined. Typically the objective will be the desired results of the order. For example, for most product manufacturers, this would be a defined quantity and quality of Supply Keeping Units (herein "SKUs") to be delivered by a certain date (i.e., Need date).

After the objective is defined, a production method or methods that meet the objective are generated at step **208**. A production method specifies the specific process steps needed to meet the objective. A production method will also identify the resources and/or resource groups (i.e., families) used for the method. The selection of resources and/or resource groups to be employed for the production method is based on the characteristics and constraints associated with each resource that was defined in step **202**. To illustrate how a production method may be generated, the following example is provided.

Referring to FIG. **3**A depicting a production method **300** for making cartons of filtered cigarettes. This production method **300** meets the objective of an order. For purposes of this example, assume that the only objective here is to produce a carton of cigarettes (in reality, an order would be typically associated with a number of other objectives, for example, need date, delivery location, and the like). In the production method **300** depicted, the method **300** begins with three nonreusable resources, tobacco **310**, filter **312** and paper **314**, being fed into a cigarette maker resource group **320**. The cigarette maker resource group **320** actually represents a production step for manufacturing cigarettes. In this case, the cigarette maker resource group **320** represents the step of making cigarettes. The cigarette makers resource group **320** is comprised of the individual cigarette makers **120**, **122**, **124** and **124** depicted in FIG. **1**. The cigarette maker resource group **320** converts the raw materials (e.g., tobacco, filter and paper) into cigarettes, which are then sent to a pack maker resource group **322**. The pack maker resource group **322** represents the production step for manufacturing cigarette packs. The pack maker resource group **322** comprises of pack makers **130** and **132** of FIG. **1**. The cigarette packs produced by the resource group for pack makers **322** is then sent to a carton maker resource group **324**. In this model, the resource group for carton maker **324** represents the production step for manufacturing cartons of cigarettes. The carton maker resource group **324** places the cigarette packs into a carton and seals the carton producing the finished goods, a carton of cigarettes **326**. Note that rather than defining specific resources, resource groups were instead used to define each step of this production method **300**. This allows greater flexibility when production routes (discussed below) are eventually generated because routes will not be restricted to using only a specific resource or resources for specific production steps. By relating each of the production steps to resource groups rather than to specific resources, the production method **300** allows greater flexibility when specific routes are eventually generated. Using resource groups rather than specific resources for specific production steps makes it possible to generate greater route variations. For example, in the above example, any of the resources or a set of resources belonging to the cigarette maker resource group **320** (which comprises of cigarette maker A, cigarette maker B, cigarette maker C and cigarette maker D) could have been used for the step of making cigarettes. Similarly, there is greater flexibility in generating routes by using a pack maker resource group **322** instead of using a specific pack maker for the step of making a pack of cigarettes.

The simplified production method **300** described above would, in actuality, also include many constraints that may exist in an actual manufacturing facility. For example, in the above method, we assume that each of the resource groups depicted produce and consume SKUs at the same rate so that all of the SKUs produced by a preceding resource group is immediately consumed by the succeeding resource group. It is assumed in the above example that the pack maker resource group **322** consumes all of the cigarettes produced by the cigarette maker resource group **320**. However, rather than consuming cigarettes on a continuous basis, suppose the pack maker resource group **322** will only consume cigarettes in batches. Thus, the pack maker resource group **322** only accepts cigarettes when enough is produced by the cigarette maker resource group **320** to make a batch. In reality, such a situation may be easily resolved by, for example, simply slowing down the production line or setting aside a place or space between the cigarette maker resource group **320** and pack maker resource group **322** to accumulate cigarettes. However, to make allowances for such discrepancies is somewhat more difficult when generating a production method. One way to deal with such discrepancies is to create a constructive resource for holding SKUs.

FIG. **3**B illustrates another production method **350**. The production method **350** is similar to the production method **300** illustrated in FIG. **3**A except that the pack maker resource group **322**A has been further defined as a "batch" pack maker resource group **322** and a constructive resource called a "holding" resource **352** is added between the cigarette maker resource group **320** and the pack maker resource group **322**A. To prevent the pack makers **130** and **132** from being able to consume anything other than batches, the pack makers **130** and **132** need to be further defined as batch consumers and producers, for example, in step **202** (FIG. **2**). This may be accomplished by creating attributes associated with each resource that defines whether each of the resources defined produce and/or consume batches of SKUs or WIPs. Further, batch size and batch production rate of SKUs or WIPs being consumed or produced must be defined for that resource.

As described above, several production methods may be generated for each objective. That is, there may be several ways to achieve the same results. The various methods generated will typically have drawbacks and/or advantages over other methods for achieving the same results. The method or methods that will be the most appropriate for a specific objective may depend on other factors, for example, user preferences.

Once production methods have been generated, one of the methods is selected for purposes of determining a feasible route at step **210**. The selection of a method may be made manually by the user, or alternatively, the system may select a method based on user-defined rules. For example, the user may create a user-defined rule that may require the system to select a method based on a preference for low cost or use of the least amount of resources, or any other user preferences.

FIG. **4**, depicts a somewhat more detailed process **400** for reviewing and selecting a feasible route or routes as depicted in step **212** of FIG. **2**. Initially at step **402**, the system generates and lists all possible routes that satisfy the production method selected at step **210**. In one embodiment, the system uses, in combination, two types of goals to select the best route. The two types of goals are scheduling goals and routing goals (described in greater detail below). At step **404** the system selects a routing goal based on user preferences. There are three types of routing goals, "fast", "best" and "fast on time". When "fast" is selected as the routing goal, the system selects the first route that is feasible. When "best" is selected as the routing goal, the system selects the route that best achieves the scheduling goal (see below). Under "best," the system will generally take longer to select a route than when the goal is set at "fast." When "fast on time" is selected as the routing goal, a blend of best and fast goals described above are sought. If this routing goal is selected, then the system checks the minimum number of routes necessary when determining which route will make the order on time. This is the goal that the user may select if the user wants to achieve a balance between using the best but most timely choice. At step **406** the system determines what type of scheduling goal is desired by the user.

In one embodiment of the process, there are five types of scheduling goals: maximizing resource utilization, just-in-time, minimize production cost, minimize cycle time and balance resource utilization (details of each type of scheduling goals is described below). Based on the combination of scheduling and routing goals selected by the user, the system selects a route at step **408**.

The specific route[s] selected is based on the combination of specific scheduling goals and routing goals selected. However, not all of the routing goals (i.e., fast, best and fast on time) are applicable to each of the five possible scheduling goals. For example, as will be discussed below, the routing goal selected will not be relevant if the scheduling goal is balance resource utilization. FIGS. **5** to **9** depict the detailed flow process **500** for selecting a route (as generally depicted in step **212** of FIG. **2**) based on the scheduling goals selected working in combination with the routing goals selected.

FIG. **5** illustrates the flow process **500** for selecting a route when the scheduling goal has been set at maximizing resource utilization. At step **501** the system generates routes based on the production method selected. At step **502** determine whether the route goal is "fast." If so, then the system selects the first route that has a scheduling opportunity at step **504**. Otherwise the system determines whether the routing goal is set at best at step **506**. If so, then the system selects the route that achieves the best route that maximizes resource utilization at step **508**. If there is a tie between two routes that are best at maximizing resource utilization then the route with the earliest scheduling opportunity is selected. If the route goal is not "best" then the system selects another production method at **510** (as generally described in step **210** in FIG. **2**) and the process of generating and selecting a route[s] begins again as indicated at **512**.

FIG. **6** depicts a detailed flow process **600** for selecting a route when the scheduling goal has been set at Just-In-Time. At step **601** the system generates routes based on the production method selected. At step **602**, determine whether the route goal is set at fast. If so, then select the first route having at least one scheduling opportunity at step **604**. At step **606**, the system determines whether there is a scheduling opportunity that results in completion exactly on the need date. If so, then the system selects that opportunity at step **608**. If not, then the system checks to see if there is a scheduling opportunity that results in completion before the need date at step **610**. If so, then the system selects that scheduling opportunity at step **612**. If not, then the system selects the scheduling opportunity that results in the earliest completion date after the need date at step **614**. If, on the other hand, the routing goal is not set at "fast" then the system determines whether the routing goal is set at "best" at step **620**. If so, then the system determines whether there

is a route with a scheduling opportunity that results in completion at the need date exactly at step **622**. If so, then the system selects that route and scheduling opportunity at step **628**. If not, then the system determines whether there is a route with a scheduling opportunity that results in completion before the need date at step **630**. If so, then the system selects that route and opportunity at step **632**. If not, then the system selects the route with the scheduling opportunity that results in the earliest completion date after the need date at step **634**. If the route goal is neither set at fast or best, then the system selects another production method at step **636** (as generally described in step **210** in FIG. **2**) and the process of generating and selecting a route[s] starts over again as indicated by **642**.

FIG. **7** depicts a detailed flow process **700** for selecting a route when the scheduling goal has been set at minimize production cost. At step **701**, generate routes based on the production method selected. At step **702**, the system determines whether the route goal is set at "fast." If so, then the system selects the first route on which there is a scheduling opportunity at step **704**. At step **706**, the system selects, from the selected first route, the scheduling opportunity that best meets the scheduling goal of minimizing production cost. If the route goal is not set at "fast" then the system determines whether the route goal is set at "fast on time" at step **708**. If so, then the system selects the first route that has a scheduling opportunity that is on time at step **710**. If not, then the system reviews all routes and selects the route that best meets the scheduling goal of minimizing production costs at step **712**. At step **714**, the system reviews all scheduling opportunities for the route selected at step **712** and selects the opportunity that best meets the scheduling goal of minimizing production cost.

FIG. **8** depicts a flow process **800** for selecting a route when the scheduling goal has been set at minimize cycle time. At step **801**, the system generates routes. At step **802**, the system determines whether the route goal is set at "best." If so, the system determines whether there are routes with scheduling opportunities that results in completion exactly on the need date at step **804**. If so, then select the first route that results in completion on the need date at step **806**. If not, then the system determines whether there are routes with a scheduling opportunity that results in a completion before the need date at step **808**. If so, then the system selects the first route having a scheduling opportunity with the latest completion date before the need date at step **810**. If not, the system selects the first route with a scheduling opportunity that results in the earliest completion date after the need date at step **812**. If, on the other hand, the route goal is not set at "best" then the system determines whether the route goal is set at "fast" at step **814**. If so, then the system selects the first route, which has a scheduling opportunity at step **816**. Otherwise, the system selects another method at step **818** (as generally indicated in step **210** of FIG. **2**) and the process for generating and selecting a route starts over again as indicated at **820**.

FIG. **9** depicts a flow process **900** for selecting a route when the scheduling goal has been set at balance resource utilization. The aim of the balance resource utilization scheduling goal is to select the route that uses the most appropriate reusable resources in terms of availability and uses the earliest scheduling opportunity in that route. At step **901**, the system generates routes based on the method selected at step **210**. At step **902**, the system calculates the average availability of each resource that is designated as balance utilization. Typically a resource may be designated as a balance utilization resource when the resource is defined. The availability of each resource designated as balance utilization may be accomplished by reviewing the calendars of each resource and determining the availability of each resource based on scheduling opportunities as indicated on the calendars. The average availability could be an average based on different time intervals such as an average over a period of a day, a week, a month or any other user defined time interval. At step **904**, the system determines qualified resources. Qualified resources are those resources designated as balance utilization needed to achieve the objective using the method selected in step **210** (see FIG. **2**). At step **906**, the system selects the route that uses the most appropriate reusable resources in terms of availability. At step **908**, the system selects the earliest scheduling opportunity for that route.

Each production method generated in step **208** (FIG. **2**) may require that more than one route be used to accomplish the objective. For example, suppose that a large customer was placed in our earlier example of the cigarette manufacturing facility. Suppose further that no single route will be able to manufacture enough cigarette cartons to fulfill the order by the need date. In such a situation, the production method selected may require that two or more routes be used to fulfill the order. Thus, the five processes described above for selecting routes based on different scheduling and routing goals may be employed multiple times for a specific method. Once the process for selecting a route and scheduling opportunities has been completed, a planned route[s] is generated. The planned route[s] will not only define the specific route[s] required for achieving the objective, but will also define the specific scheduling opportunities for each targeted resources being used for achieving the objective.

Referring back to FIG. **2**, based on the selected route, assignments are generated for each resource targeted for use according to the selected route at step **214**. An assignment is an order or a reservation for a specific resource reserving the resource for use during a specified time period (i.e., time slot). In a preferred embodiment, the system will review the assignments to determine whether any "product wheels" are being violated at step **216**. A product wheel is a constraint placed on specific resources preventing or at least warning the user that an undesirable "transition" is being scheduled. A transition is the process that a resource must undertake between succeeding assignments. For example, manufacturing equipment must typically undergo certain modifications when the type of products being produced by the equipment is switched. The cost and time needed to undergo these transitions is typically unwelcome. However, the impact of transitions may be minimized by carefully selecting the sequence of products being produced by a particular resource. Thus, some companies prefer to schedule product families in "campaigns." which are preferred sequences of products scheduled over a set period of time. For example, a company may want to schedule a product family of SKUs of a particular grade or color, followed by a different product family.

"Product wheels" are a particular kind of sequencing problem in which orders for products must follow a prescribed set of transitions from product to product within a group of resources. The preferred or allowable transitions from one SKU to another on a resource may be important factors in a good schedule.

To illustrate the problem of product wheels, the following example is provided. Suppose an oil production manufacturer produces different kinds of oil. The oils are differentiated by their viscosities or grades. To manufacture the

different grades of oil, the manufacturer uses an additive in its oil, that, when added in different percentages, create oils of differing grades. To optimize its manufacturing facility, the manufacturer prefers that certain machines transition from making one grade to another. Depending on the grades being produced, it can be time-consuming (and therefore more expensive) to switch a machine from making one grade of oil to another, because the machine may require cleaning during transition. The more the grades differ, the more cleaning is required.

To illustrate the point, we refer to FIG. **10**A showing four grades of oil and the transitions that are allowed when scheduling an oil mixer (not shown) that manufactures the different grades of oil. In this illustration, the oil mixer may produce oil of various grades, A, B, C and D **1010**, **1012**, **1014** and **1016**. The mixer may transition from producing grade A oil **1010** to either grade B oil **1012** or grade C oil **1014**. The mixer may then transition from producing grade B or C oil **1012** and **1014** to a grade D oil **1016**. Finally, the mixer may transition from producing grade D oil **1016** to grade A oil **1010**. A product wheel that would constrain the oil mixer in a manner described above would constrain all of the transitions illustrated above.

To illustrate how a product wheel may constrain a particular resource, we now refer to FIG. **10**B. FIG. **10**B depicts a chart **1020** that shows the various transitions of an exemplary product wheel associated with the transitions allowed for an oil mixer illustrated in FIG. **10**A. Rows **1030** to **1038** are the transitions that make up a product wheel called "Oil Mixer 1." Row **1040** is a transition for another product wheel associated with another resource called "Packager 1. " These transitions in rows **1030** to **1038** are called allowed or permitted transitions. Thus, each transition is associated with a particular product wheel as indicated in column **1050** (in this case either Oil Mixer 1 or Packager 1). The SKUs shown in column **1052** are the initial SKUs being manufactured by the oil mixer prior to transition, and the SKUs shown in column **1054** are the ending SKUs after transition. Thus, the product wheel "oil mixer 1" will only permit those transitions listed in the chart **1020** of FIG. **10**B.

Other constraints may also be included in a product wheel. For example, a cost constraint, as indicated in column **1056**, prevents oil mixer 1 from exceeding certain costs for certain transitions. In row **30**, the transition from oil grade A to oil grade B for oil mixer 1 is limited to the cost of $30.00. Each product wheel may be assigned to one or more resource. Although a product wheel may be assigned to more than one resource, a resource may only be associated with one product wheel at any given time.

FIG. **11** depicts a flow process **1100** for accommodating product wheel constraints as generally depicted at step **216** in FIG. **2**. Once the assignments have been generated in step **214** (see FIG. **2**), the system checks to see if any of the targeted resources are associated with any product wheel, and if so, whether any of the product wheels are violated by one or more of the assignments, at step **1102**. If no product wheels have been violated then the assignments are scheduled at step **1104**. If, on the other hand, the system determines that one or more product wheels have been violated then the system may alert the user of this fact at step **1106**. At step **1108**, the system determines whether to override the violated product wheel[s]. If so, then the assignments are scheduled at step **1110**. If, on the other hand, it is decided that no product wheel should be violated, then the new assignments are cancelled at step **1112**. At step **1114**, determine whether there are other scheduling opportunities for the selected route. If so, then at step **1116**, generate new assignments for the scheduling opportunities and return to step **1102**. If not, then select another route per step **212** of FIG. **2**, step **1118**.

According to another preferred embodiment of the present invention, the flow process of FIG. **2** may also include the step of packing out excess WIP at step **218**. The step of packing out excess WIP performs the step of consuming any excess WIPs generated by an order. Although shown as part of the overall flow process of FIG. **2**, the step of packing out excess WIP may be implemented completely independently from the flow process.

For various reasons, such as reducing storage costs, it is highly desirable to minimize excess WIPs kept in inventory. Excess WIPs may exist in inventory for many reasons, such as the conflicting capacity levels of different resources. For example, a resource may only be able to manufacture SKUs in batches rather than continuously. However, suppose an order is scheduled which requires less then the minimum number of SKUs that a resource may be able to produce. As a result, excess SKUs, which may be either WIPs or finished goods, may remain.

Referring to FIG. **12** depicting a flow process **1200** for packing out excess WIPs. At step **1202**, the system determines that excess WIPs exists. This may be accomplished in a number of ways. For example, when scheduling an order the system may make a determination that excess WIPs will be generated as a result of scheduling an order by examining the production method, routes and the scheduling opportunities used for that order. Alternatively, the system may store in the database **172**, information relating to all SKUs, both finished goods and WIPs, being generated and/or stored in inventory. The system determines when excess WIPs exists by continuously or by periodically monitoring information stored in the database. If excess WIP is detected then at step **1204** the system reviews stored orders to determine whether any of the stored orders can use the excess WIP during the manufacturing process of fulfilling the orders. This may be accomplished in a number of ways. For example, the system may define all of the SKUs generated (both finished goods and WIPs) by at least two attributes. One attribute could identify those SKUs that should be packed out when excess units exist. The second attribute may define the finished goods that are associated with the SKU. That is, the finished goods that consumes that SKU during the manufacturing process of the finished goods. By using such an attribute to define the SKUs, the system can identify a stored order that uses the excess SKU simply by reviewing the finished goods associated with each of the orders stored by the system. Alternatively, another method that the system may use in determining which of the stored orders may be able to use the excess WIP is to further define the orders by defining the intermediate SKUs required for producing the finished goods. This may be accomplished by at least two ways, by initially identifying all the needed intermediate SKUs for an order when the order is initially created and stored or by using the system to generate production methods for an order when the order is first received and defining the order by the intermediate SKUs required for the production methods generated. Once an order that uses the excess WIP has been identified, the order is scheduled at step **1206**. At step **1206** the system determines whether all of the excess WIP have been consumed. If excess WIP still remains then the system goes back to step **1204** to determine and schedule orders that consumes the excess WIP.

Although the steps illustrated in the flow process depicted in FIG. **2** are generally shown to be in a particular order or sequence, there is no strict requirement that each step must

be performed in the order generally illustrated in FIG. **2**. Further, certain steps may be optional, for example, the packing out step **216** may be optional.

According to another embodiment of the present invention, the system may employ block scheduling. When block scheduling is employed, users can reserve capacity of selected resources to particular product family during specified time intervals. Further, the activities or assignments scheduled during the reserved time interval will be in a particular order similar to the concept of product wheels. Together with the product wheel feature, this feature helps to schedule orders in a way, which results in a more efficient and/or cost effective manner.

To reserve a resource, a block calendar is created and assigned to the resource being reserved. The block calendar includes blocks of time, each having a set start and end times, and a product family assigned to it. When an order for one of the SKUs in the product family is actually scheduled, the assignments for it will be scheduled on the resource during only those blocks allocated to it on the resource's block schedule. A product family is a group of products or SKUs that may be grouped together because they have some common attribute. The foregoing description of the preferred embodiments of the present invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. It will be apparent to those of ordinary skill in the art that various modifications and variations can be made to the system and method embodied by the present invention without departing from the spirit or scope of the invention. Thus, it is intended that the present invention covers the modifications and variations of this invention provided that they come within the scope of any claims and their equivalents.

We claim:

1. A system for generating and selecting a planning route and selecting scheduling opportunities according to the selected route, comprising:

means for defining resources;

means for receiving an order having an objective;

means for generating a production method based on said objective and said defined resources;

means for generating routes based on said production method; and

means for selecting one of said routes and scheduling opportunities based on a scheduling goal and a routing goal.

2. The system of claim **1**, wherein said scheduling goal is selected from the group consisting of maximizing resource utilization, just-in-time, minimize production cost, minimize cycle time and balance resource utilization.

3. The system of claim **2**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

4. The system of claim **1**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

5. The system of claim **1**, further comprising a means for creating a product wheel.

6. The system of claim **5**, wherein said means for creating a product wheel comprises a means for defining allowed transitions for one of said resources.

7. The system of claim **6**, further comprising a means for reviewing said selected route and scheduling opportunities to determine whether said selected route and scheduling opportunities violates said product wheel and selecting another one of said routes generated if said product wheel has been violated.

8. The system of claim **1**, further comprising a means for packing out excess work-in-process.

9. The system of claim **8**, wherein said means for packing out excess work-in-process comprises the step of identifying said excess work-in-process.

10. The system of claim **9**, wherein said means of packing out excess work-in-process comprises a means for identifying orders that consume said excess work-in-process.

11. The system of claim **10**, wherein said means for packing out excess work-in-process comprises a means for scheduling said identified orders.

12. The system of claim **11**, further comprising a means for assigning a first attribute and a second attribute to a SKU.

13. The system of claim **12**, wherein said first attribute indicates whether said SKU needs to be packed out when said SKU is present in excess.

14. The system of claim **13**, wherein said second attribute indicates a finished good that is associated with said SKU.

15. The system of claim **1**, wherein said means for defining resources comprises a means for defining said resources by identifying starting and ending SKUs, production rates and resource families.

16. The system of claim **15**, wherein said means for generating a production method based on said objective and said defined resources further based on said resource families.

17. A system for generating and selecting a planning route and selecting scheduling opportunities according to the selected route, comprising:

a database storing manufacturing data and receiving and storing an order, wherein said order having an objective;

a modeling module which defines resources;

a production method module which generates a production method based on said objective and said defined resources;

a routing module which generates routes based on said production method; and

a scheduling manager module which selects one of said routes and scheduling opportunities based on a scheduling goal and a routing goal.

18. The system of claim **17**, wherein said scheduling goal is selected from the group consisting of maximizing resource utilization, just-in-time, minimize production cost, minimize cycle time and balance resource utilization.

19. The system of claim **18**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

20. The system of claim **17**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

21. The system of claim **17**, further comprising a product wheel module which creates a product wheel.

22. The system of claim **21**, wherein said product wheel module further defines allowed transitions for one of said resources.

23. The system of claim **22**, where in said product wheel module further reviews said selected route and scheduling opportunities to determine whether said selected route and scheduling opportunities violates said product wheel and selects another one of said routes generated if said product wheel has been violated.

24. The system of claim **17**, further comprising a pack out module which packs out excess work-in-process.

25. The system of claim **24**, wherein said pack out module packs out excess work-in-process by identifying said excess work-in-process.

26. The system of claim **25**, wherein said packing out module further identifies orders that consume said excess work-in-process.

27. The system of claim **26**, wherein said packing out module further schedules said identified orders.

28. The system of claim **27**, wherein said scheduling manager module assigns a first attribute and a second attribute to a SKU.

29. The system of claim **28**, wherein said first attribute indicates whether said SKU needs to be packed out when said SKU is present in excess.

30. The system of claim **29**, wherein said second attribute indicates a finished good that is associated with said SKU.

31. The system of claim **17**, wherein said modeling module which defines resources is by identifying starting and ending SKUs, production rates and resource families associated with said resources.

32. The system of claim **31**, wherein said production method module which generates a production method based on said objective and said defined resources is further based on said resource families.

33. A method for generating and selecting a planning route and selecting scheduling opportunities according to the selected route, comprising the steps:

defining resources;

receiving an order having an objective;

generating a production method based on said objective and said defined resources;

generating routes based on said production method; and

selecting one of said routes and scheduling opportunities based on a scheduling goal and a routing goal.

34. The method of claim **33**, wherein said scheduling goal is selected from the group consisting of maximizing resource utilization, just-in-time, minimize production cost, minimize cycle time and balance resource utilization.

35. The method of claim **34**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

36. The method of claim **33**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

37. The method of claim **33**, further comprising the step of creating a product wheel.

38. The method of claim **37**, wherein said step of creating a product wheel comprises the step of defining allowed transitions for one of said resources.

39. The method of claim **38**, further comprising the steps of reviewing said selected route and scheduling opportunities to determine whether said selected route and scheduling opportunities violates said product wheel and selecting another one of said routes generated if said product wheel has been violated.

40. The method of claim **33**, further comprising the step of packing out excess work-in-process.

41. The method of claim **40**, wherein said step of packing out excess work-in-process comprises the step of identifying said excess work-in-process.

42. The method of claim **41**, wherein said step of packing out excess work-in-process comprises the step of identifying orders that consume said excess work-in-process.

43. The method of claim **42**, wherein said step of packing out excess work-in-process comprises the step of scheduling said identified orders.

44. The method of claim **43**, further comprising the step of assigning a first attribute and a second attribute to a SKU.

45. The method of claim **44**, wherein said first attribute indicates whether said SKU needs to be packed out when said SKU is present in excess.

46. The method of claim **45**, wherein said second attribute indicates a finished good that is associated with said SKU.

47. The method of claim **33**, wherein said step of defining resources comprises the step of defining said resources by identifying starting and ending SKUs, production rates and resource families.

48. The method of claim **47**, wherein said step of generating a production method based on said objective and said defined resources further based on said resource families.

49. A program storage device readable by a machine, tangibly embodying a program of instructions executable by a machine to perform the steps of generating and selecting a planning route and selecting scheduling opportunities according to the selected route, comprising the steps:

defining resources;

receiving an order having an objective;

generating a production method based on said objective and said defined resources;

generating routes based on said production method; and

selecting one of said routes and scheduling opportunities based on a scheduling goal and a routing goal.

50. The program storage device of claim **49**, wherein said scheduling goal is selected from the group consisting of maximizing resource utilization, just-in-time, minimize production cost, minimize cycle time and balance resource utilization.

51. The program storage device of claim **50**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

52. The program storage device of claim **49**, wherein said routing goal is selected from the group consisting of fast, best and fast on time.

53. The program storage device of **49**, further comprising the step of creating a product wheel.

54. The program storage device of claim **53**, wherein said step of creating a product wheel comprises the step of defining allowed transitions for one of said resources.

55. The program storage device of claim **54**, further comprising the steps of reviewing said selected route and scheduling opportunities to determine whether said selected route and scheduling opportunities violates said product wheel and selecting another one of said routes generated if said product wheel has been violated.

56. The program storage device of claim **49**, further comprising the step of packing out excess work-in-process.

57. The program storage device of claim **56**, wherein said step of packing out excess work-in-process comprises the step of identifying said excess work-in-process.

58. The program storage device of claim **57**, wherein said step of packing out excess work-in-process comprises the step of identifying orders that consume said excess work-in-process.

59. The program storage device of claim **58**, wherein said step of packing out excess work-in-process comprises the step of scheduling said identified orders.

60. The program storage device of claim **59**, further comprising the step of assigning a first attribute and a second attribute to a SKU.

61. The program storage device of claim **60**, wherein said first attribute indicates whether said SKU needs to be packed out when said SKU is present in excess.

62. The program storage device of claim **61**, wherein said second attribute indicates a finished good that is associated with said SKU.

63. The program storage device of claim **49**, wherein said step of defining resources comprises the step of defining said resources by identifying starting and ending SKUs, production rates and resource families.

64. The program storage device of claim **63**, wherein said step of generating a production method based on said objective and said defined resources further based on said resource families.

* * * * *

EXHIBIT 2

US007050874B1

(12) **United States Patent**
Tenorio

(10) **Patent No.: US 7,050,874 B1**
(45) **Date of Patent: May 23, 2006**

(54) **SOURCING OF BILLS OF MATERIALS**

(75) Inventor: **Manoel Tenorio**, Mountain View, CA (US)

(73) Assignee: **i2 Technologies US, Inc.**, Dallas, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 624 days.

(21) Appl. No.: **09/976,791**

(22) Filed: **Oct. 12, 2001**

(51) **Int. Cl.**
G06F 17/00 (2006.01)
(52) **U.S. Cl.** **700/107**; 716/1
(58) **Field of Classification Search** 700/107; 716/1
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,204,821	A *	4/1993	Inui et al.	700/106
5,515,269	A	5/1996	Willis et al.	364/401
5,630,070	A *	5/1997	Dietrich et al.	705/8
5,835,910	A *	11/1998	Kavanagh et al.	707/103 R
5,970,476	A *	10/1999	Fahey	705/28
6,119,102	A *	9/2000	Rush et al.	705/29
6,199,068	B1 *	3/2001	Carpenter	707/100
6,226,618	B1 *	5/2001	Downs et al.	705/1
6,493,679	B1 *	12/2002	Rappaport et al.	705/29
6,584,370	B1 *	6/2003	Denton et al.	700/107
6,625,616	B1 *	9/2003	Dragon et al.	707/104.1
6,650,954	B1 *	11/2003	Zulpa et al.	700/107
6,819,967	B1 *	11/2004	Ballas et al.	700/107
6,901,304	B1 *	5/2005	Swan et al.	700/115

* cited by examiner

Primary Examiner—Andrew Joseph Rudy
(74) *Attorney, Agent, or Firm*—Brian E. Harris; James E. Walton

(57) **ABSTRACT**

A bill of materials (BOM) sourcing system includes one or more data storage locations that store BOM sourcing criteria. The system also includes a sourcing engine that receives a BOM that includes a plurality of part identifiers. The sourcing engine identifies one or more part identifiers included in the BOM and searches supplier data to obtain pricing information associated with the identified parts. Furthermore, the sourcing engine receives sourcing criteria and determines which supplier should be used to supply each identified part based on the pricing information and the sourcing criteria. In addition, the system includes a transaction execution module that receives from the sourcing engine a part identifier for each identified part and the identity of the supplier to be used to supply each identified part. The transaction execution module generates one or more transaction documents based on this information and communicates the one or more transaction documents to the associated suppliers.

12 Claims, 1 Drawing Sheet

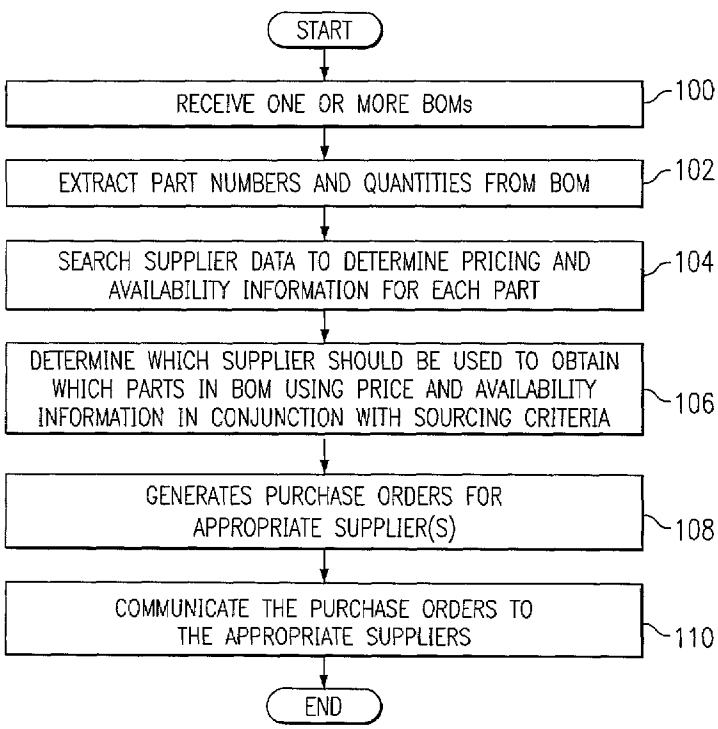


FIG. 1

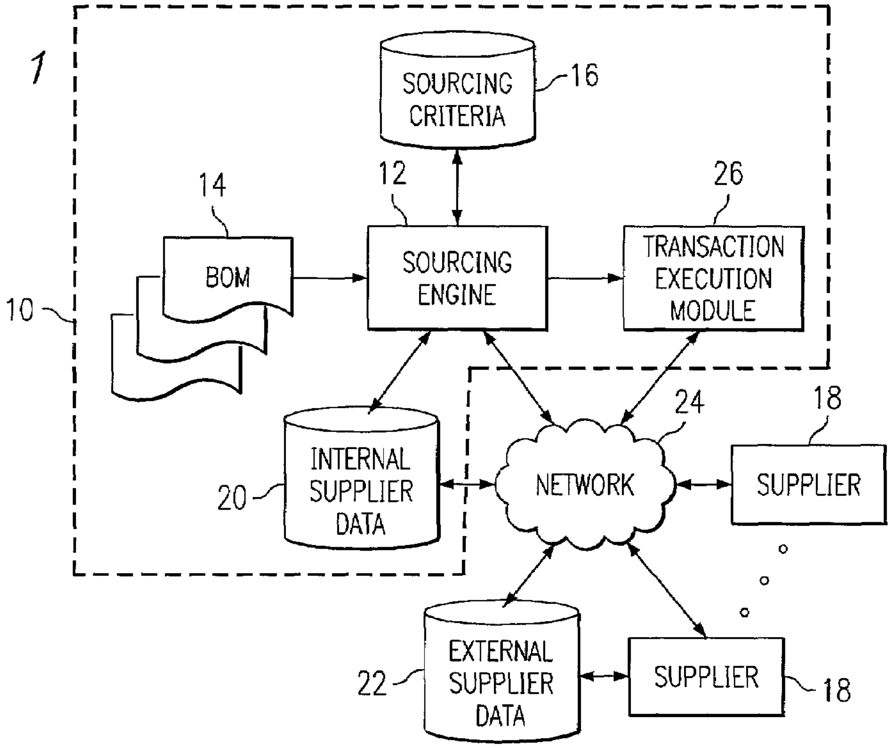
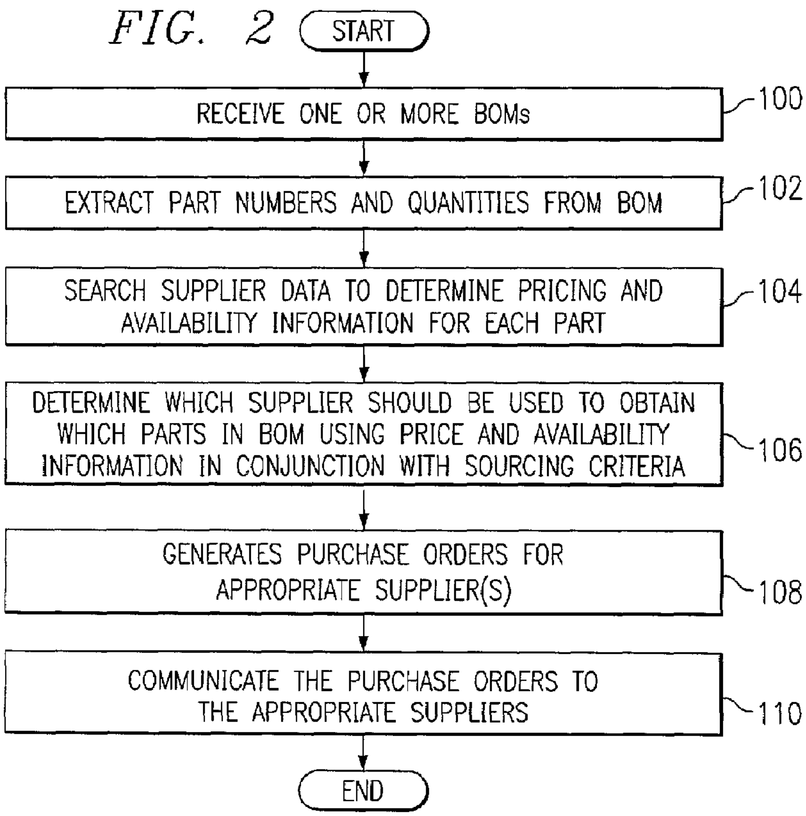

SOURCING CRITERIA
16
14
12
26
BOM
SOURCING ENGINE
TRANSACTION EXECUTION MODULE
10
24
18
INTERNAL SUPPLIER DATA
20
NETWORK
SUPPLIER
EXTERNAL SUPPLIER DATA
22
SUPPLIER
18

FIG. 2


START
RECEIVE ONE OR MORE BOMs
100
EXTRACT PART NUMBERS AND QUANTITIES FROM BOM
102
SEARCH SUPPLIER DATA TO DETERMINE PRICING AND AVAILABILITY INFORMATION FOR EACH PART
104
DETERMINE WHICH SUPPLIER SHOULD BE USED TO OBTAIN WHICH PARTS IN BOM USING PRICE AND AVAILABILITY INFORMATION IN CONJUNCTION WITH SOURCING CRITERIA
106
GENERATES PURCHASE ORDERS FOR APPROPRIATE SUPPLIER(S)
108
COMMUNICATE THE PURCHASE ORDERS TO THE APPROPRIATE SUPPLIERS
110
END

SOURCING OF BILLS OF MATERIALS

TECHNICAL FIELD OF THE INVENTION

This invention relates to the field of electronic commerce transactions, and more particularly to improved sourcing of bills of materials.

BACKGROUND OF THE INVENTION

Bills of materials (BOMs) are typically used to identify the various parts that are included in a particular product. Therefore, a BOM is a useful tool for conveying the information about what parts are needed to manufacture or otherwise create a product. BOMs are created by a BOM originator (such as an original equipment manufacturer) and then communicated in whole or in part to one or more suppliers (for example, as part of a request for quote). The suppliers then determine which parts are included in the product (or a portion of the product for which the supplier is responsible) and quote a price for each part or for a collection of parts to the BOM originator. Each supplier may also provide additional information, such as part availability and lead time, as appropriate. The BOM originator collects the various responses from the suppliers, analyzes the responses, determines which suppliers to use based on the responses, and communicates purchase orders or other transaction documents to the selected suppliers. This process may be referred to as "sourcing" a BOM.

One disadvantage with the process described above is that it inefficient for the BOM originator to have to collect and organize the various responses from the suppliers (which may be in a variety of formats) and to determine which suppliers to use based on a number of decision criteria. For example, in addition to pricing considerations, the BOM originator may have to review various contracts with suppliers to determine if the contracts have an effect on which suppliers are selected (for example, if the BOM originator has agreed to do a certain amount of business with a particular supplier). Furthermore, the process described above may take too long for certain circumstances since the BOM originator has to wait on the suppliers for responses and then has to analyze the responses before orders can be placed.

SUMMARY OF THE INVENTION

According to the present invention, disadvantages and problems associated with previous techniques for sourcing bills of materials have been substantially reduced or eliminated.

According to one embodiment of the present invention, a bill of materials (BOM) sourcing system includes one or more data storage locations that store BOM sourcing criteria. The system also includes a sourcing engine that receives a BOM that includes a plurality of part identifiers. The sourcing engine identifies one or more part identifiers included in the BOM and searches supplier data associated with one or more suppliers of parts to obtain pricing information associated with the identified parts. Furthermore, the sourcing engine receives sourcing criteria from one or more of the data storage locations and determines which supplier should be used to supply each identified part based on the pricing information and the sourcing criteria. In addition, the system includes a transaction execution module that receives from the sourcing engine a part identifier for each identified part and the identity of the supplier to be used to supply each

identified part. The transaction execution module generates one or more transaction documents that identify one or more parts and include a request that a supplier provide the one or more parts and communicates the one or more transaction documents to the associated suppliers.

Particular embodiments of the present invention provide one or more technical advantages. For example, certain embodiments provide a bill of materials (BOM) sourcing system that can source one or more BOMs without having to communicate the BOMs to suppliers. This system has access to the supplier data so that pricing and other appropriate information may be discovered and the system includes the intelligence needed to select from the various suppliers based on the supplier data and appropriate selection crtieria. Therefore, BOMs may be sourced in a much quicker and more efficient manner than with previous BOM sourcing techniques.

Other technical advantages may be readily apparent to those skilled in the art from the figures, description and claims included herein.

BRIEF DESCRIPTION OF THE DRAWINGS

To provide a more complete understanding of the present invention and the features and advantages thereof, reference is made to the following description taken in conjunction with the accompanying drawings, in which:

FIG. **1** illustrates an example bill of materials sourcing system; and

FIG. **2** illustrates an example method of sourcing a bill of materials.

DESCRIPTION OF EXAMPLE EMBODIMENTS

FIG. **1** illustrates an example bill of materials (BOM) sourcing system **10**. A BOM is a document or other data file that often includes a list of parts (or components) and sub-parts that are included in one or more products. For example, a BOM for a bicycle might include a list of parts such as a frame, wheels, handle bars, pedals, sprocket, and chain. Some of these parts may also have one or more associated sub-parts included in the BOM. For example, a wheel part or "assembly" may include a tire sub-part and a rim sub-part. Each part and/or sub-part in the BOM may have an associated part number, name, description and/or other appropriate part identifier and a required quantity of each part. Therefore, the BOM may be used to identify every part needed to make a product and the quantity of each part that is needed. Although an example BOM is described, the term "bill of materials" (or "BOM") is meant to include any list or other compilation of part or part information (including, but not limited to part numbers, part names, part descriptions, and manufacturer names) and is not limited to any particular formats of such information.

Since BOMs may be used to identify all the parts of a product, a BOM is a useful method of conveying the information about what parts are needed to manufacture or otherwise create a product. As an example only, an original equipment manufacturer (OEM) may generate a BOM that identifies a number of parts that are included in a particular product sold by the OEM. The product or portions of the product may be made for the OEM by one or more suppliers. In the past, the OEM would typically communicate the BOM to the suppliers that are to provide the parts and/or manufacture the product or portions of the product. For example, the BOM may be included as part of a request for quote (RFQ) associated with the product. The suppliers have

then used the BOM to provide a quote to the OEM regarding the cost of providing one or more parts, or producing the product or the relevant portions of the product. This process may be referred to as "sourcing" a BOM.

In the sourcing process, the suppliers take the part number or other associated information from the BOM and use these part numbers to identify the parts needed and to determining pricing and other relevant information (such as available supply) for these parts. The OEM or other originator of the BOM then receives the pricing and other relevant information from the suppliers and determines how to fill the parts requirements (in other words, how to "source" the BOM). The BOM originator may consider a number of different factors besides price in determining how to source the BOM. As examples only, the BOM originator may have specific approved suppliers from which it may source BOMs and/or the BOM originator may have limitations on the number of suppliers that may be used to source a BOM. Once the BOM originator has determined how to source the BOM, the BOM originator may then communicate purchase orders or other appropriate documents to the suppliers.

System **10** provides an improved technique for sourcing BOMs. The components of system **10** allow BOMs to be sourced by system **10** without having to communicate the BOMs to suppliers. Furthermore, system **10** can enforce appropriate rules when sourcing the BOMs. The components of system **10** may be implemented as software and/or hardware associated with one or more computers in one or more locations. For example, system **10** may be associated with an OEM or other BOM originator or may be implemented at an electronic marketplace or other networked electronic trading site. System **10** includes a sourcing engine **12** that receives one or more BOMs **14**, analyzes BOMs **14**, and sources BOMs **14** according to sourcing information, restrictions, and rules (collectively referred to as "sourcing criteria" **16**) stored in one or more data storage locations. Sourcing criteria **16** may include information identifying approved suppliers **18**, information regarding contractual requirements or limitations from contracts with suppliers **18** that may affect the sourcing of BOMs **14**, various other sourcing rules, and any other appropriate criteria for use in sourcing BOMs **14**.

Any appropriate sourcing rules may be used. As examples only, these rules may include a rule specifying that a BOM **14** should be sourced at the lowest cost, a rule specifying that the BOM originator should only buy a certain amount of parts or products from a particular supplier **18**, and/or a rule specifying that the number of suppliers **18** used to supply parts in a BOM **14** should be minimized. Each such rule may be enforced with respect to the other rules. For example some or all of the rules may be hard rules or constraints that may not be violated. Alternatively or in addition, some or all of the rules may be soft rules that may be violated. Such violations may incur penalties (possibly relative to the degree the rule is violated) that encourage the soft rules to be enforced, but that allow them be violated if such a violation produces a sourcing that is optimal or preferred.

As described above, one factor that may (and typically is) considered in sourcing a BOM **14** is the cost associated with sourcing the BOM **14** or portions of the BOM **14** from one or more suppliers **18**. For example, as described above, sourcing engine **12** may implement a sourcing rule that attempts to minimize the cost of acquiring the parts or products in a BOM **14**. To allow such cost and pricing determinations to be made, sourcing engine **12** may have access to one or more sources supplier data (such as pricing and available quantity information). This supplier data may

include internal supplier data **20** stored in one or more data storage locations associated with system **10** and/or external supplier data **22** stored in one or more data storage locations external to system **10**. Suppliers **18** may communicate internal supplier data **20** to system **10** on a periodic basis, as needed or requested by system **10**, and/or in any other appropriate manner. Internal supplier data **20** may be communicated to system **10** using a communication network **24**. In an example embodiment, network **24** includes the Internet and any appropriate local area networks (LANs), metropolitan area networks (MANs), or wide area networks (WANs) coupling system **10** to suppliers **18**. External supplier data **22** may be maintained by suppliers **18** at one or more locations external to system **20** and made available to users using network **24** or any in other appropriate manner. System **10** may access external supplier data **22** as needed or desired to make sourcing determinations.

In operation, sourcing engine **12** receives one or more BOMs **14** from an appropriate source. For example, sourcing engine **12** may receive BOMs **14** from a local or remote BOM generation tool or from another appropriate component associated with a BOM originator. Sourcing engine **12** may then extract the part number or other identifier of each part from BOM **14** and use these part numbers or other identifiers to formulate one or more search queries of internal supplier data **20** and/or external supplier data **20**. Through such search queries or other appropriate techniques of identifying appropriate supplier data, sourcing engine **12** may determine pricing and availability information associated with one or more suppliers **18** of each part listed in BOM **14**. Sourcing engine **12** may then use this price and availability information in conjunction with sourcing criteria **16** to determine which supplier **18** should be used to obtain which parts or products included in BOM **14**.

System **10** also may include a transaction execution module **26** that can implement the sourcing decisions made by sourcing engine **12**. Transaction execution module **26** may communicate purchase orders or other similar transaction documents to suppliers **18** using network **24** or other appropriate communication techniques. For example, the transaction documents may be communicated using electronic data interchange techniques, electronic mail, web-based communications (such as hypertext mark-up language or extensible mark-up language), or faxes. These transaction documents reflect the sourcing determinations made by sourcing engine **12**. For example, transaction execution module **26** may communicate a purchase order to each supplier **18** that sourcing engine **12** determined was to be used to provide one or more parts from a BOM **14**. For instance, the purchase order may indicate the part number and description, the quantity needed, and the price that was quoted from the internal or external source of supplier data.

FIG. **2** illustrates an example method of sourcing a BOM **14** using system **10**. The method begins at step **100** where sourcing engine **12** receives one or more BOMs **14** from an appropriate source. At step **102**, sourcing engine **12** extracts the part number or other identifier for each part in BOM **14** and uses these part numbers at step **104** to search internal supplier data **20** and/or external supplier data and determine pricing and availability information for one or more suppliers **18** for each part listed in BOM **14**. At step **106**, sourcing engine **12** uses this price and availability information in conjunction with sourcing criteria **16** to determine which supplier **18** should be used to obtain which parts or products included in BOM **14**. If appropriate, these determinations may be communicated to one or more sources for approval (for example, sourcing engine **12** may communicate selected

determinations based on appropriate criteria, such as the size or amount of a contemplated purchase). Alternatively, the determinations may be communicated directly to transaction execution module **26**. Transaction execution module **26** receives these determinations (either with or without approval) and generates purchase orders or other similar transaction documents at step **108** for the supplier(s) **18** that were determined to be used as a source for the parts. At step **110**, transaction execution module **26** uses network **24** or other appropriate communication resources to communicate the purchase orders to the appropriate suppliers **18**, and the method ends. In such a manner, one or more BOMs **14** may be sourced by system without having to communicate the BOMs to suppliers (such as in an RFQ process). Therefore, BOMs **14** may be sourced in a much quicker and more efficient manner than with previous BOM sourcing techniques.

Although the present invention has been described with several embodiments, numerous changes, substitutions, variations, alterations, and modifications may be suggested to one skilled in the art, and it is intended that the invention encompass all such changes, substitutions, variations, alterations, and modifications as fall within the spirit and scope of the appended claims.

What is claimed is:

1. A bill of materials (BOM) sourcing system, comprising:
- one or more data storage locations operable to store BOM sourcing criteria;
- a sourcing engine operable to:
  - receive a BOM including a plurality of part identifiers;
  - identify one or more part identifiers included in the BOM;
  - search supplier data associated with one or more suppliers of parts to obtain pricing information associated with the identified parts;
  - receive the sourcing criteria from the one or more data storage locations; and
  - determine which supplier should be used to supply each identified part based on the pricing information and the sourcing criteria; and
- a transaction execution module operable to:
  - receive from the sourcing engine a part identifier for each identified part and the identity of the supplier to be used to supply each identified part;
  - generate one or more transaction documents each identifying one or more parts and including a request that a supplier provide the one or more parts; and
  - communicate the one or more transaction documents to the associated suppliers.

2. The system of claim **1**, wherein the sourcing engine is further operable to:
- identify in the BOM a required quantity of each identified part;
- search the supplier data for availability information associated with the identified parts; and
- use the availability information and required quantity when determining which supplier should be used to supply each identified part.

3. The system of claim **1**, wherein one or more of the data storage locations are operable to store supplier data locally at the bill of materials sourcing system, the one or more storage locations coupled to a communication network and operable to receive updated supplier data using the network.

4. The system of claim **1**, wherein the supplier data is stored in one or more data storage locations external to the bill of materials sourcing system, the sourcing engine operable to access the supplier data using a communication network.

5. The system of claim **1**, wherein the sourcing criteria comprise information identifying approved suppliers of one or more parts, the suppliers used to supply each identified part being limited to the approved suppliers.

6. The system of claim **1**, wherein the sourcing criteria comprise contractual requirements with one or more suppliers that affect the sourcing of the BOM.

7. The system of claim **1**, wherein the sourcing criteria comprise a rule specifying that the BOM should be sourced at the lowest cost.

8. The system of claim **1**, wherein the sourcing criteria comprise a rule limiting the amount of parts obtained from a particular supplier.

9. The system of claim **1**, wherein the sourcing criteria comprise a rule specifying that the total number of suppliers used to supply the identified parts should be minimized.

10. The system of claim **1**, wherein the transaction documents further identify the pricing information obtained from the supplier data for each part.

11. The system of claim **1**, wherein the transaction execution module is operable to communicate the one or more transaction documents using a communication technique selected from the group consisting of electronic mail, fax, web-based communications, and electronic data interchange (EDI).

12. A bill of materials (BOM) sourcing system, comprising:
- means for receiving a BOM, the BOM including a plurality of part identifiers;
- means for identifying one or more part identifiers included in the BOM;
- means for searching supplier data associated with one or more suppliers of parts to obtain pricing information associated with the identified parts;
- means for receiving sourcing criteria;
- means for determining which supplier should be used to supply each identified part based on the pricing information and the sourcing criteria;
- means for generating one or more transaction documents each identifying one or more parts and including a request that the determined supplier provide the one or more parts; and
- means for communicating the one or more transaction documents to the associated suppliers.

* * * * *

EXHIBIT 3

US007574383B1

(12) **United States Patent**
Parasnis et al.

(10) **Patent No.: US 7,574,383 B1**
(45) **Date of Patent: Aug. 11, 2009**

(54) **SYSTEM AND METHOD FOR PROVIDING DISTRIBUTED INVENTORY MANAGEMENT**

(75) Inventors: **Abhay Vinayak Parasnis**, Coppell, TX (US); **Ashish Bharara**, Dallas, TX (US)

(73) Assignee: **i2 Technologies US, Inc.**, Dallas, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 716 days.

(21) Appl. No.: **10/120,571**

(22) Filed: **Apr. 10, 2002**

Related U.S. Application Data

(60) Provisional application No. 60/283,448, filed on Apr. 11, 2001.

(51) **Int. Cl.**
G06Q 30/00 (2006.01)

(52) **U.S. Cl.** **705/28**; 705/8; 705/22

(58) **Field of Classification Search** 705/22–28, 705/30, 8
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,608,621 A * 3/1997 Caveney et al. 705/10
5,712,989 A * 1/1998 Johnson et al. 705/28
5,765,143 A * 6/1998 Sheldon et al. 705/28
6,081,789 A * 6/2000 Purcell 705/37
6,341,271 B1 * 1/2002 Salvo et al. 705/28
6,374,254 B1 * 4/2002 Cochran et al. 707/102
6,421,676 B1 * 7/2002 Krishnamurthy et al. 707/102

OTHER PUBLICATIONS

U.S. Appl. No. 10/119,990, entitled "*Intelligent Fulfillment Agents*,", 46 pages, filed Apr. 10, 2002.

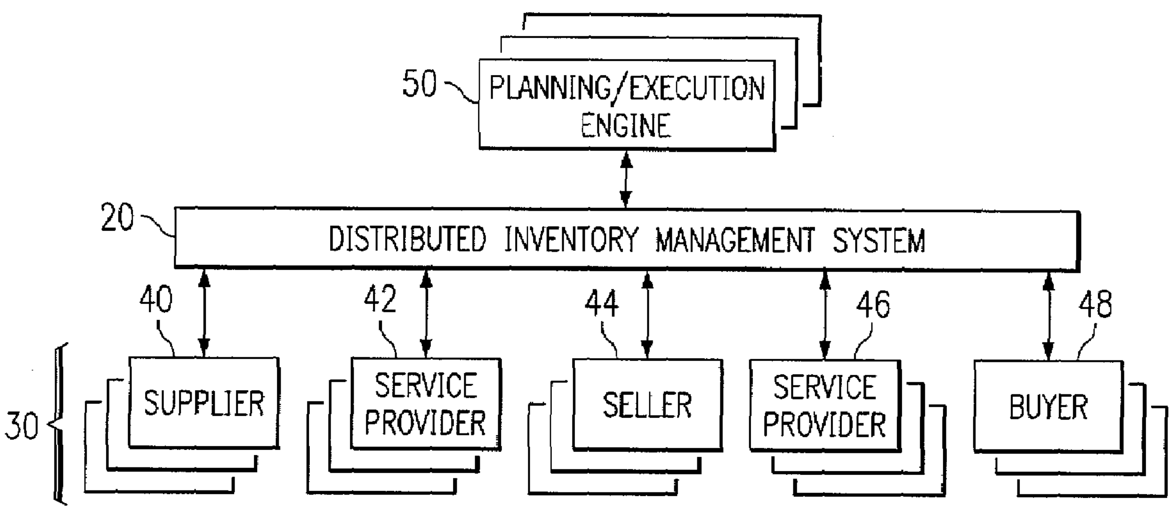
* cited by examiner

Primary Examiner—F. Zeender
Assistant Examiner—Garcia Ade
(74) *Attorney, Agent, or Firm*—Booth Udall, PLC; Steven J. Laureanti

(57) **ABSTRACT**

A method for distributed inventory management includes receiving information regarding a number of participants in a value chain and information regarding one or more items relevant in the value chain. The method also includes modeling relationships between two or more of the participants based on the received information and modeling the one or more items based on the received information. Furthermore, the method includes receiving inventory data from the participants relating to the one or more items, processing the inventory data based on the models of the relationships and the items to generate inventory information related to one or more of the participants, and communicating the generated inventory information to one or more of the participants. The inventory information at least partially includes information regarding inventory data of participants other than those participants to which the inventory information is communicated.

30 Claims, 6 Drawing Sheets



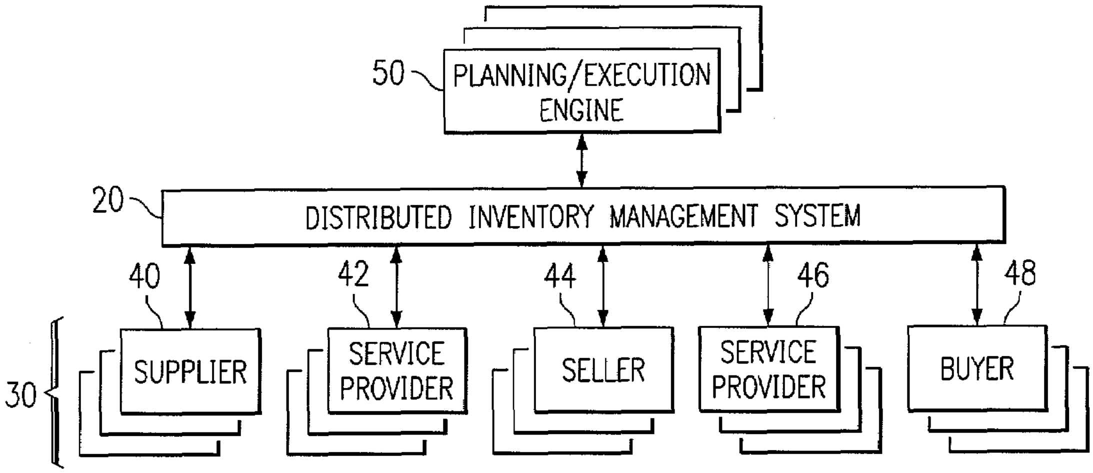

50
PLANNING/EXECUTION ENGINE
20
DISTRIBUTED INVENTORY MANAGEMENT SYSTEM
40
42
44
46
48
30
SUPPLIER
SERVICE PROVIDER
SELLER
SERVICE PROVIDER
BUYER

FIG. 1

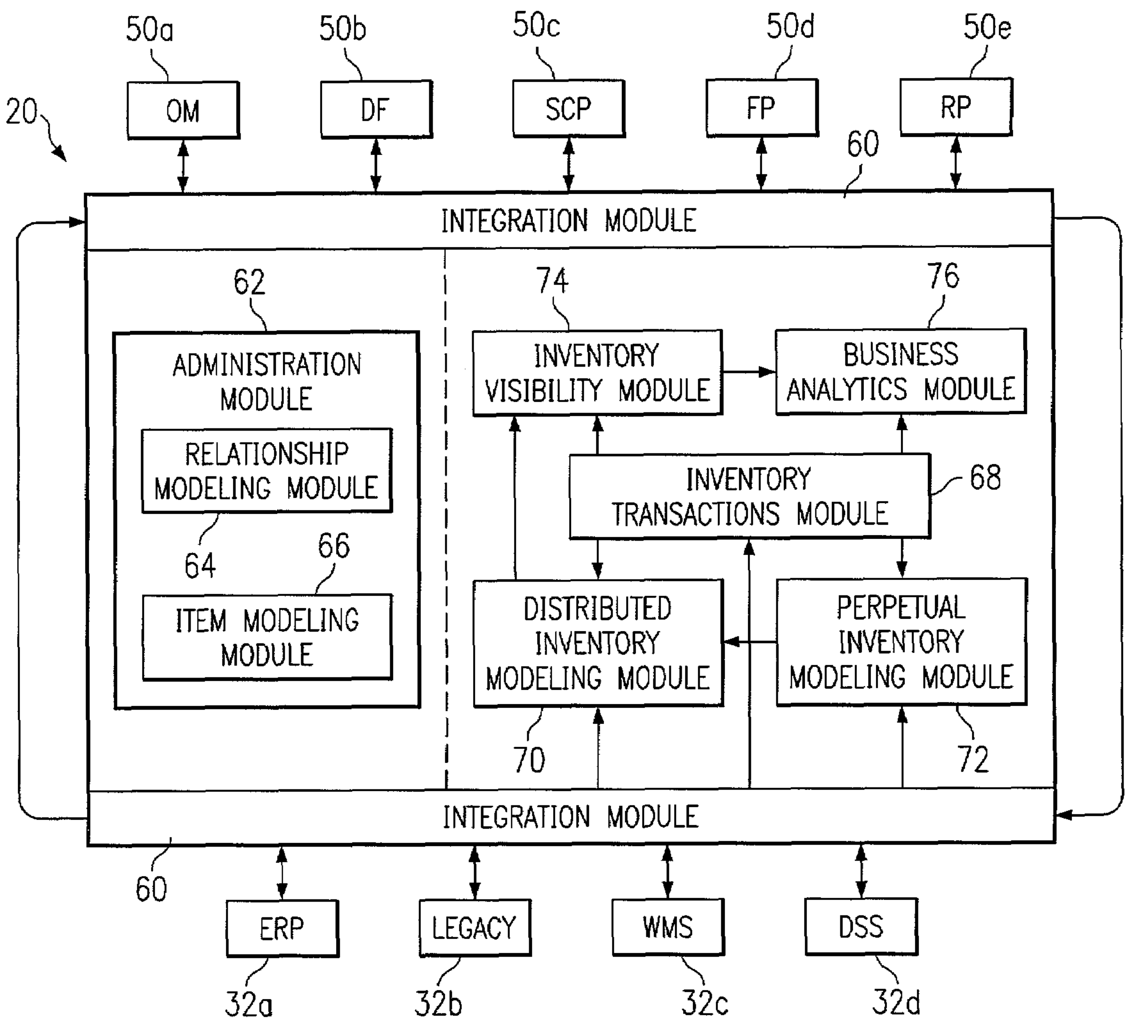

50a
50b
50c
50d
50e
20
OM
DF
SCP
FP
RP
60
INTEGRATION MODULE
62
74
76
ADMINISTRATION MODULE
RELATIONSHIP MODELING MODULE
64
66
ITEM MODELING MODULE
INVENTORY VISIBILITY MODULE
BUSINESS ANALYTICS MODULE
INVENTORY TRANSACTIONS MODULE
68
DISTRIBUTED INVENTORY MODELING MODULE
PERPETUAL INVENTORY MODELING MODULE
70
72
INTEGRATION MODULE
60
ERP
LEGACY
WMS
DSS
32a
32b
32c
32d

FIG. 2

FIG. 3

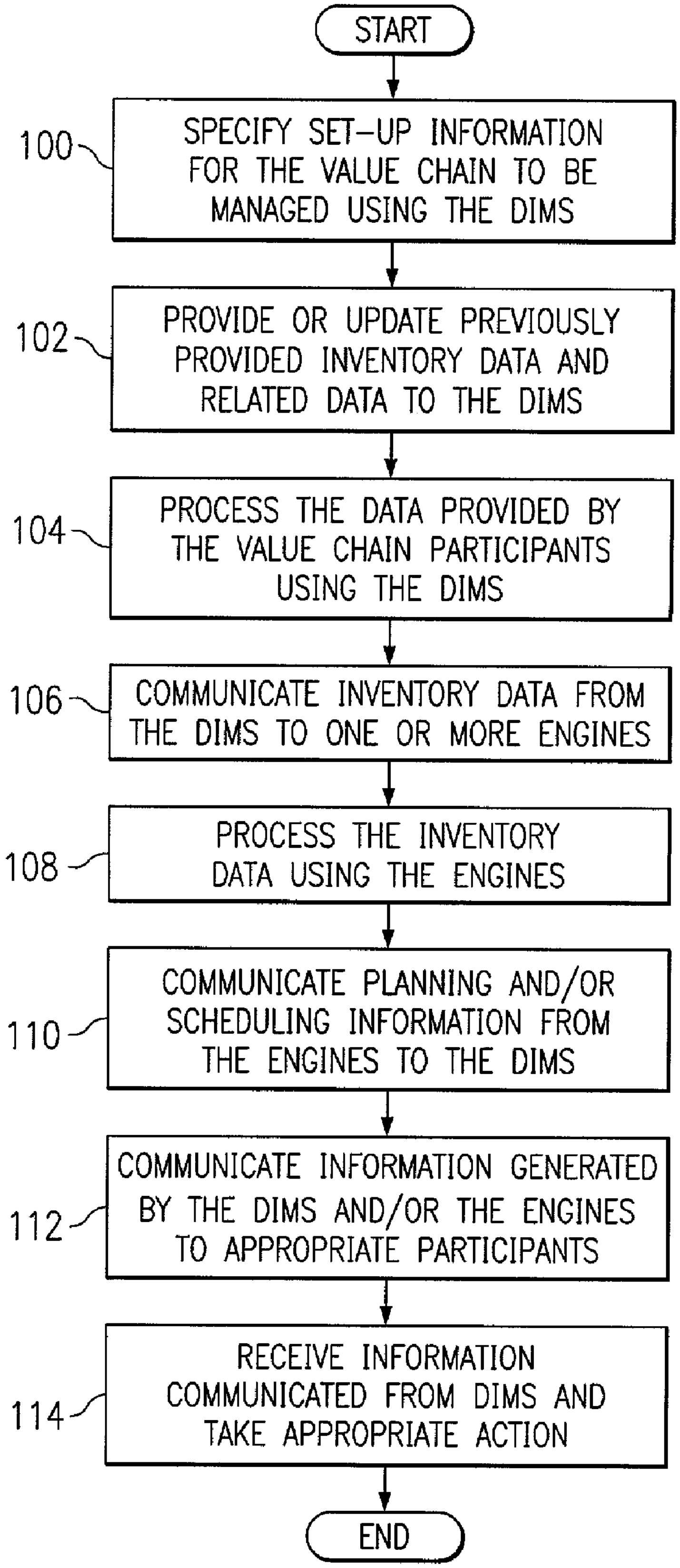

START
100
SPECIFY SET-UP INFORMATION FOR THE VALUE CHAIN TO BE MANAGED USING THE DIMS
102
PROVIDE OR UPDATE PREVIOUSLY PROVIDED INVENTORY DATA AND RELATED DATA TO THE DIMS
104
PROCESS THE DATA PROVIDED BY THE VALUE CHAIN PARTICIPANTS USING THE DIMS
106
COMMUNICATE INVENTORY DATA FROM THE DIMS TO ONE OR MORE ENGINES
108
PROCESS THE INVENTORY DATA USING THE ENGINES
110
COMMUNICATE PLANNING AND/OR SCHEDULING INFORMATION FROM THE ENGINES TO THE DIMS
112
COMMUNICATE INFORMATION GENERATED BY THE DIMS AND/OR THE ENGINES TO APPROPRIATE PARTICIPANTS
114
RECEIVE INFORMATION COMMUNICATED FROM DIMS AND TAKE APPROPRIATE ACTION
END

FIG. 4

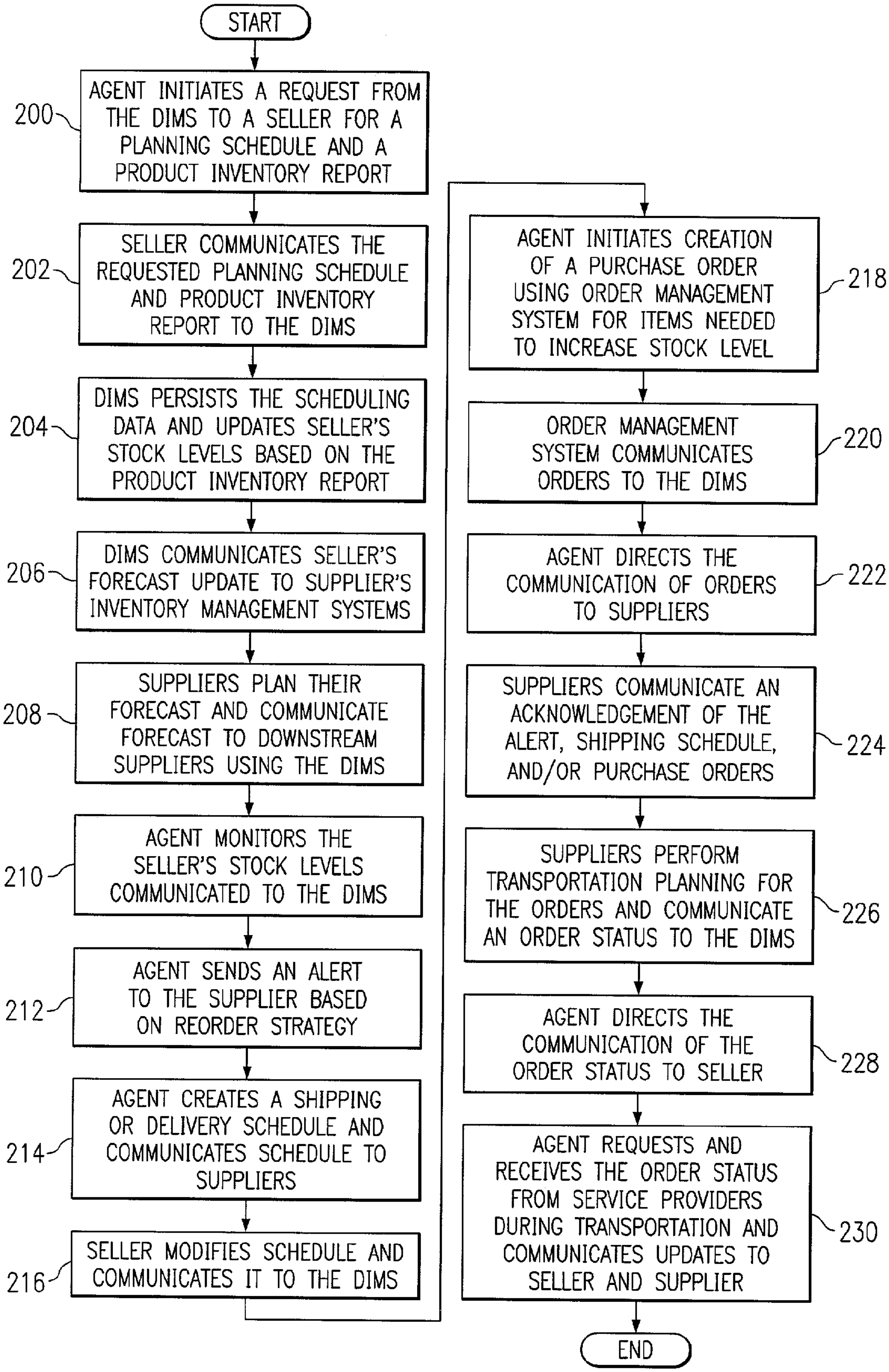

START
AGENT INITIATES A REQUEST FROM THE DIMS TO A SELLER FOR A PLANNING SCHEDULE AND A PRODUCT INVENTORY REPORT
200
SELLER COMMUNICATES THE REQUESTED PLANNING SCHEDULE AND PRODUCT INVENTORY REPORT TO THE DIMS
202
DIMS PERSISTS THE SCHEDULING DATA AND UPDATES SELLER'S STOCK LEVELS BASED ON THE PRODUCT INVENTORY REPORT
204
DIMS COMMUNICATES SELLER'S FORECAST UPDATE TO SUPPLIER'S INVENTORY MANAGEMENT SYSTEMS
206
SUPPLIERS PLAN THEIR FORECAST AND COMMUNICATE FORECAST TO DOWNSTREAM SUPPLIERS USING THE DIMS
208
AGENT MONITORS THE SELLER'S STOCK LEVELS COMMUNICATED TO THE DIMS
210
AGENT SENDS AN ALERT TO THE SUPPLIER BASED ON REORDER STRATEGY
212
AGENT CREATES A SHIPPING OR DELIVERY SCHEDULE AND COMMUNICATES SCHEDULE TO SUPPLIERS
214
SELLER MODIFIES SCHEDULE AND COMMUNICATES IT TO THE DIMS
216
AGENT INITIATES CREATION OF A PURCHASE ORDER USING ORDER MANAGEMENT SYSTEM FOR ITEMS NEEDED TO INCREASE STOCK LEVEL
218
ORDER MANAGEMENT SYSTEM COMMUNICATES ORDERS TO THE DIMS
220
AGENT DIRECTS THE COMMUNICATION OF ORDERS TO SUPPLIERS
222
SUPPLIERS COMMUNICATE AN ACKNOWLEDGEMENT OF THE ALERT, SHIPPING SCHEDULE, AND/OR PURCHASE ORDERS
224
SUPPLIERS PERFORM TRANSPORTATION PLANNING FOR THE ORDERS AND COMMUNICATE AN ORDER STATUS TO THE DIMS
226
AGENT DIRECTS THE COMMUNICATION OF THE ORDER STATUS TO SELLER
228
AGENT REQUESTS AND RECEIVES THE ORDER STATUS FROM SERVICE PROVIDERS DURING TRANSPORTATION AND COMMUNICATES UPDATES TO SELLER AND SUPPLIER
230
END

FIG. 5A

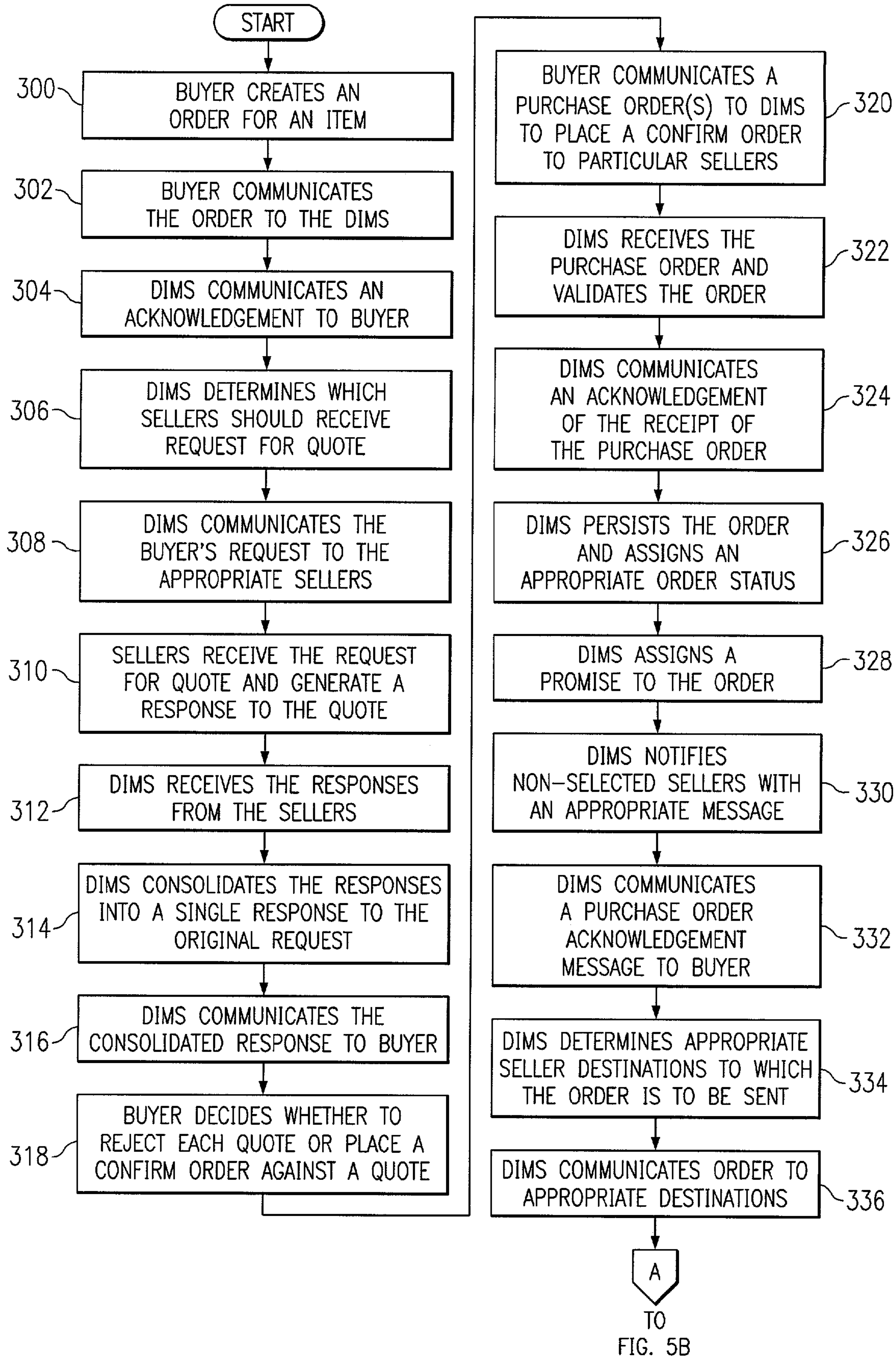

START
300 BUYER CREATES AN ORDER FOR AN ITEM
302 BUYER COMMUNICATES THE ORDER TO THE DIMS
304 DIMS COMMUNICATES AN ACKNOWLEDGEMENT TO BUYER
306 DIMS DETERMINES WHICH SELLERS SHOULD RECEIVE REQUEST FOR QUOTE
308 DIMS COMMUNICATES THE BUYER'S REQUEST TO THE APPROPRIATE SELLERS
310 SELLERS RECEIVE THE REQUEST FOR QUOTE AND GENERATE A RESPONSE TO THE QUOTE
312 DIMS RECEIVES THE RESPONSES FROM THE SELLERS
314 DIMS CONSOLIDATES THE RESPONSES INTO A SINGLE RESPONSE TO THE ORIGINAL REQUEST
316 DIMS COMMUNICATES THE CONSOLIDATED RESPONSE TO BUYER
318 BUYER DECIDES WHETHER TO REJECT EACH QUOTE OR PLACE A CONFIRM ORDER AGAINST A QUOTE
320 BUYER COMMUNICATES A PURCHASE ORDER(S) TO DIMS TO PLACE A CONFIRM ORDER TO PARTICULAR SELLERS
322 DIMS RECEIVES THE PURCHASE ORDER AND VALIDATES THE ORDER
324 DIMS COMMUNICATES AN ACKNOWLEDGEMENT OF THE RECEIPT OF THE PURCHASE ORDER
326 DIMS PERSISTS THE ORDER AND ASSIGNS AN APPROPRIATE ORDER STATUS
328 DIMS ASSIGNS A PROMISE TO THE ORDER
330 DIMS NOTIFIES NON-SELECTED SELLERS WITH AN APPROPRIATE MESSAGE
332 DIMS COMMUNICATES A PURCHASE ORDER ACKNOWLEDGEMENT MESSAGE TO BUYER
334 DIMS DETERMINES APPROPRIATE SELLER DESTINATIONS TO WHICH THE ORDER IS TO BE SENT
336 DIMS COMMUNICATES ORDER TO APPROPRIATE DESTINATIONS
A
TO FIG. 5B

FIG. 5B

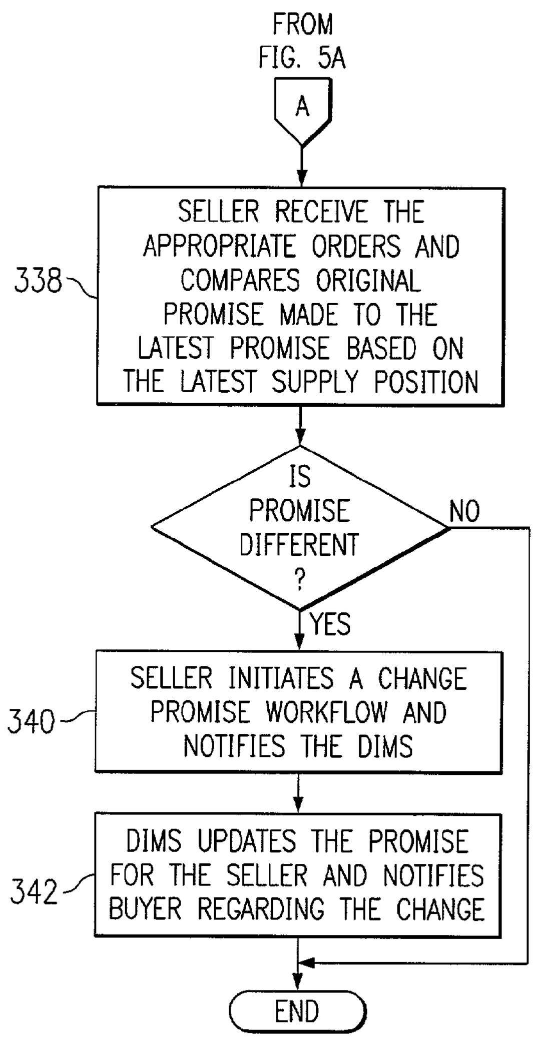
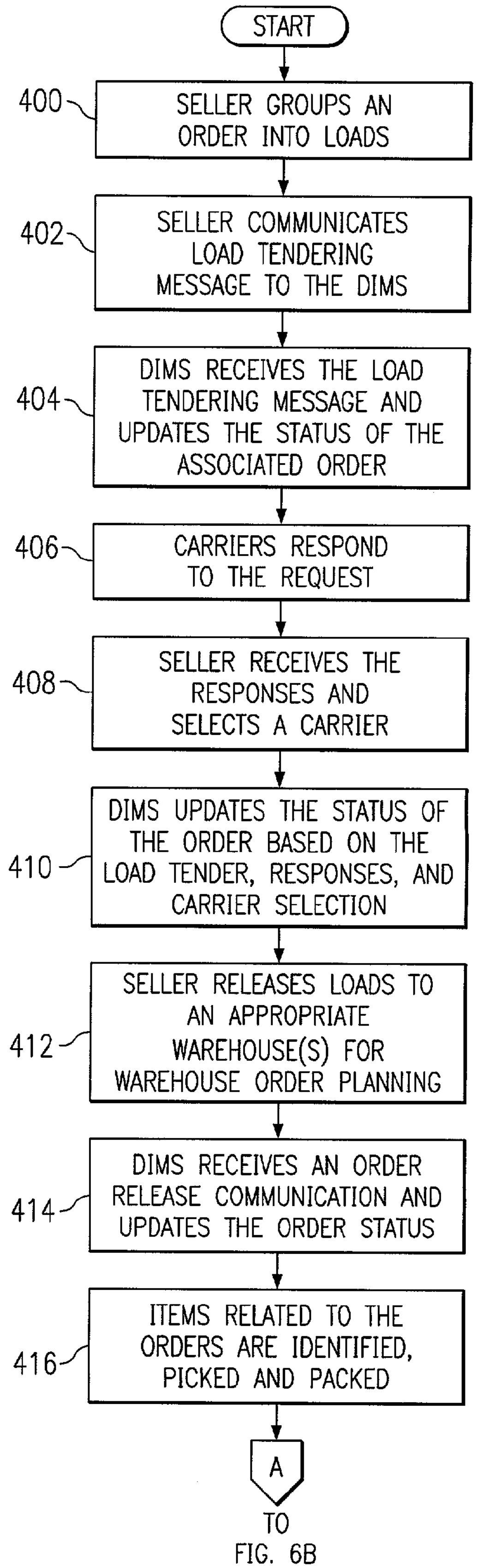

FROM
FIG. 5A
A
338
SELLER RECEIVE THE APPROPRIATE ORDERS AND COMPARES ORIGINAL PROMISE MADE TO THE LATEST PROMISE BASED ON THE LATEST SUPPLY POSITION
IS PROMISE DIFFERENT ?
NO
YES
340
SELLER INITIATES A CHANGE PROMISE WORKFLOW AND NOTIFIES THE DIMS
342
DIMS UPDATES THE PROMISE FOR THE SELLER AND NOTIFIES BUYER REGARDING THE CHANGE
END

FIG. 6A


START
400
SELLER GROUPS AN ORDER INTO LOADS
402
SELLER COMMUNICATES LOAD TENDERING MESSAGE TO THE DIMS
404
DIMS RECEIVES THE LOAD TENDERING MESSAGE AND UPDATES THE STATUS OF THE ASSOCIATED ORDER
406
CARRIERS RESPOND TO THE REQUEST
408
SELLER RECEIVES THE RESPONSES AND SELECTS A CARRIER
410
DIMS UPDATES THE STATUS OF THE ORDER BASED ON THE LOAD TENDER, RESPONSES, AND CARRIER SELECTION
412
SELLER RELEASES LOADS TO AN APPROPRIATE WAREHOUSE(S) FOR WAREHOUSE ORDER PLANNING
414
DIMS RECEIVES AN ORDER RELEASE COMMUNICATION AND UPDATES THE ORDER STATUS
416
ITEMS RELATED TO THE ORDERS ARE IDENTIFIED, PICKED AND PACKED
A
TO
FIG. 6B

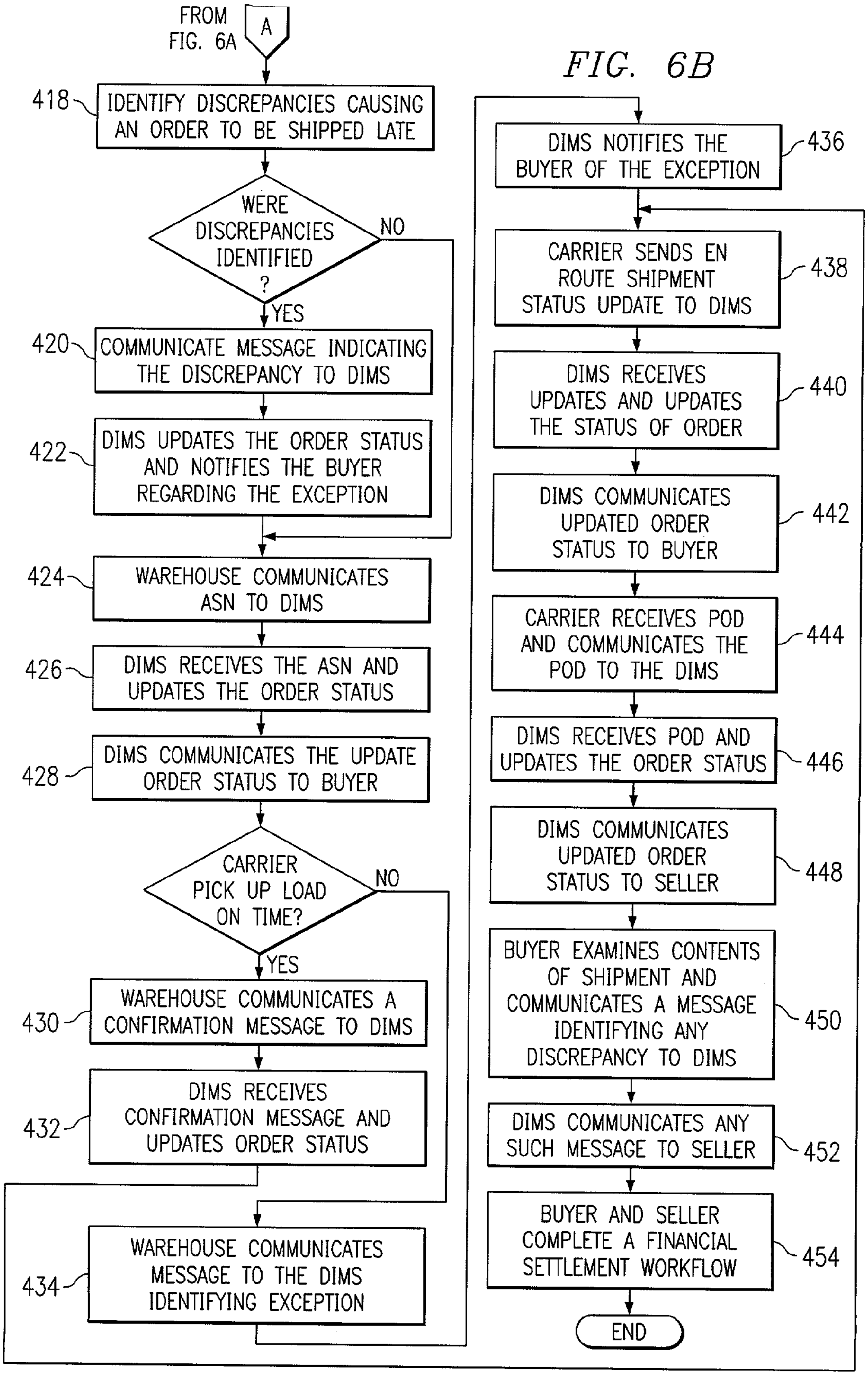

FROM FIG. 6A
A
418
IDENTIFY DISCREPANCIES CAUSING AN ORDER TO BE SHIPPED LATE
WERE DISCREPANCIES IDENTIFIED ?
NO
YES
420
COMMUNICATE MESSAGE INDICATING THE DISCREPANCY TO DIMS
422
DIMS UPDATES THE ORDER STATUS AND NOTIFIES THE BUYER REGARDING THE EXCEPTION
424
WAREHOUSE COMMUNICATES ASN TO DIMS
426
DIMS RECEIVES THE ASN AND UPDATES THE ORDER STATUS
428
DIMS COMMUNICATES THE UPDATE ORDER STATUS TO BUYER
CARRIER PICK UP LOAD ON TIME?
NO
YES
430
WAREHOUSE COMMUNICATES A CONFIRMATION MESSAGE TO DIMS
432
DIMS RECEIVES CONFIRMATION MESSAGE AND UPDATES ORDER STATUS
434
WAREHOUSE COMMUNICATES MESSAGE TO THE DIMS IDENTIFYING EXCEPTION
DIMS NOTIFIES THE BUYER OF THE EXCEPTION
436
CARRIER SENDS EN ROUTE SHIPMENT STATUS UPDATE TO DIMS
438
DIMS RECEIVES UPDATES AND UPDATES THE STATUS OF ORDER
440
DIMS COMMUNICATES UPDATED ORDER STATUS TO BUYER
442
CARRIER RECEIVES POD AND COMMUNICATES THE POD TO THE DIMS
444
DIMS RECEIVES POD AND UPDATES THE ORDER STATUS
446
DIMS COMMUNICATES UPDATED ORDER STATUS TO SELLER
448
BUYER EXAMINES CONTENTS OF SHIPMENT AND COMMUNICATES A MESSAGE IDENTIFYING ANY DISCREPANCY TO DIMS
450
DIMS COMMUNICATES ANY SUCH MESSAGE TO SELLER
452
BUYER AND SELLER COMPLETE A FINANCIAL SETTLEMENT WORKFLOW
454
END

FIG. 6B

SYSTEM AND METHOD FOR PROVIDING DISTRIBUTED INVENTORY MANAGEMENT

RELATED APPLICATIONS

This application claims the benefit under 35 U.S.C. § 119 (e) of U.S. Provisional Application Ser. No. 60/283,448, filed Apr. 11, 2001 and entitled System Incorporating Distributed Inventory Backplane, Intelligent Collaborative Fulfillment Agents, And Distributed Order Fulfillment Concepts, Singly Or In Any Combination.

TECHNICAL FIELD OF THE INVENTION

This invention relates generally to the field of business management and more particularly to distributed inventory management.

BACKGROUND OF THE INVENTION

Various inventory replenishment processes are employed by businesses to fulfill orders and manage inventory. Although business use various types of replenishment processes (which are often customized for the business), some of the most commonly used processes are Vendor-Managed Inventory (VMI) programs, Supplier-Managed Inventory (SMI) programs and Just in Time (JIT) programs. In a VMI program, a vendor of products takes over the responsibility of managing the inventory of certain products for a given customer. Depending on the situation, the vendor might receive data such as forecasted demand, product consumption rates, inventory positions from the customer, and other inventory-related data. The vendor may also or alternatively be responsible for generating part of the data on the customers behalf. The vendor then uses the received and/or generated data to determine how much of the product to replenish and when such replenishment should occur.

In an SMI program, a seller negotiates with a sub-set of its suppliers to create supplier hubs dedicated to that seller (and which may be managed by a third party). As part of the negotiation, the third party and the suppliers agree to particular service levels at a hub, such as speed of delivery of replenishments of inventory and minimum and maximum stock levels. On a periodic basis throughout a given day, the seller may communicate with the third party provider to move material from the hub to the plant (a JIT arrangement). On a weekly or monthly basis, the seller may send a forecast to the suppliers. The suppliers are then responsible for maintaining inventory within the negotiated levels at the hub using this daily product movement and forecast information.

Although these various replenishment processes provide adequate inventory management, problems arise when multiple entities are involved in a value chain and especially when these multiple entities carry out different types of replenishment programs between one another.

SUMMARY OF THE INVENTION

According to the present invention, disadvantages and problems associated with previous business management systems have been substantially reduced or eliminated.

According to one embodiment of the present invention, a method for distributed inventory management includes receiving information regarding a number of participants in a value chain and information regarding one or more items relevant in the value chain. The method also includes modeling relationships between two or more of the participants based on the received information and modeling the one or more items based on the received information. Furthermore, the method includes receiving inventory data from the participants relating to the one or more items, processing the inventory data based on the models of the relationships and the items to generate inventory information related to one or more of the participants, and communicating the generated inventory information to one or more of the participants. The inventory information at least partially includes information regarding inventory data of participants other than those participants to which the inventory information is communicated.

Particular embodiments of the present invention may provide one or more technical advantages. For example, certain embodiments of the present invention real-time visibility into the inventory information associated with distributed participants in multiple tiers of a value chain. Particular embodiments also integrate these participants and their inventory information with various decision support planning and execution systems and provide business analysis with features such as monitoring of key performance measures (such as order fill rates, shipment variance, and inventory turns), reporting, and exception and alert management. Certain embodiments also provide integration into planning, order management and warehouse management systems of value chain participants.

Embodiments of the present invention may also provide improvements over existing systems, such as reduced capital blocked in inventory (resulting in improved return on assets), reduced warehouse and handling costs, reduced cost of goods sold, increased inventory turns, reduced logistics costs, decreased exposure to price protection liabilities, and increased customer satisfaction.

Other important technical advantages are readily apparent to those skilled in the art from the figures, descriptions and claims included herein.

BRIEF DESCRIPTION OF THE DRAWINGS

To provide a more complete understanding of the present invention and the features and advantages thereof, reference is made to the following description taken in conjunction with the accompanying drawings, in which:

FIG. **1** illustrates an example distributed inventory management system for managing the inventory of one or more participants in a value chain;

FIG. **2** illustrates an example distributed inventory management system in further detail;

FIG. **3** illustrates an example method for replenishment planning and execution using a distributed inventory management system;

FIG. **4** illustrates an example inventory workflow that may be implemented and automated using agents;

FIGS. **5**A and **5**B illustrate an example distributed order fulfillment method; and

FIGS. **6**A and **6**B illustrates an example distributed order execution method.

DESCRIPTION OF EXAMPLE EMBODIMENTS

FIG. **1** illustrates an example distributed inventory management system (DIMS) **20** for managing the inventory of one or more participants **30** in a value chain. DIMS **20** couples inventory data resources of a number of different participants **30** in a value chain with one or more planning and/or execution engines **50** that use the inventory data to plan and execute inventory-related business decisions. Participants **30** may

include one or more suppliers **40** of components used to manufacture or otherwise create products, one or more sellers **44** that manufacture or otherwise create the products (for example, a seller **44** may be an original equipment manufacturer), and one or more buyers **48** that purchase the products from sellers **44**. Participants **30** may also include one or more service providers **42** and **46** that serve as intermediaries between suppliers **40** or buyers **48** and sellers **44**. As examples only, service providers **42** may include third party logistics (3PL) providers, contract sellers, or distributors, and service providers **46** may include 3PL providers, value-added resellers (VARs), or distributors.

It should be noted that each participant **30** may act as a "buyer" or a "seller" with respect to another entity in the value chain and the terms used above to name each participant **30** should not be construed as limiting the roles of participants **30**. Furthermore, participants may include one or more sub-organizations that carry out the various roles. As an example only, a seller **44** may be a manufacturer that includes a buying sub-organization that interacts with suppliers **40** (either directly or through a service provider **42**) to obtain components, a manufacturing sub-organization that manufactures one or more products using the components, and a selling sub-organization that interacts with buyers **48** (either directly or through a service provider **46**) to sell the manufactured products. Participants **30** may alternatively or additionally include any other entities participating in a value chain. The present invention also contemplates that some or all of participants **30** may be associated with the same entity (for example, multiple participants **30** may be different divisions of a company).

In general, DIMS **20** brokers pertinent data between participants **30** and engines **50** to propagate inventory planning and execution information between participants. For example, participants **30** may communicate inventory data to DIMS **20** and DIMS **20** may process this data and communicate inventory information from the processed data back to one or more participants **30**. DIMS **20** may also or alternatively communicate this data from participants **30** to one or more engines **50**. Engines **50** may then communicate inventory planning and execution information back to DIMS **20**, which may then make this information available to some or all of participants **30**. Although DIMS **20** and engines **50** are illustrated in FIG. **1** as being separate from participants **30**, it should be understood that some or all of DIMS **20** may be associated with one or more of participants **30**. For example, some or all modules of DIMS **20**, as will be described below, may be associated with a seller **44** and the suppliers **40**, service providers **42** and **46**, and/or the buyers **48** who wish to share inventory planning and execution information amongst one another may communicate inventory data to and receive inventory information from the components of DIMS **20** associated with the seller **44**. As an alternative, DIMS **20** may be implemented independently from participants **30** (for example, in an e-commerce marketplace or other node of a trading network). In a similar manner, each engine **50** may be associated with a particular participant **30** or may be implemented independently from participants **30**. As another alternative, a DIMS **20** may exist at each of a number of locations (either associated with a participant **30** or a third party) and/or the components of an individual DIMS **20** may be distributed amongst multiple locations.

Participants **30** may each operate one or more computer systems at one or more locations. These systems of participants **30** may interact with DIMS **20** autonomously or according to input from one or more users associated with a participant **30**. DIMS **20**, the systems of participants **30**, and engines

50 may be coupled to one another using one or more local area networks (LANs), metropolitan area networks (MANs), wide area networks (WANs), a portion of the global computer network known as the Internet, or any other appropriate wireline, wireless, or other links. Participants **30** and DIMS **20** may be arranged and communicate with one another according to a hub-and-spoke, peer-to-peer, or any other suitable architecture.

The use of DIMS **20** for inventory management provides numerous advantages over previous techniques, which have inherent inefficiencies that are caused by a lack of connectivity, timely communication, and visibility between the various participants **30**. Although current techniques for implementing replenishment processes (such as VMI, SMI, and JIT) are suitable for the needs of some businesses, these techniques include several disadvantages. In many value chains, different replenishment programs and policies are used in different tiers of the value chain. For example, a JIT program may be used between a seller **44** and a service provider **42** and a separate SMI policy may be used between the service provider **42** and a supplier **40**. Because of the disconnect between the JIT policy used between the seller **44** and the service provider **42** and the SMI policy used between the service provider **42** and the supplier **40**, inefficiencies are created in the value chain. Furthermore, it is also common in value chains for a participant **30**, such as an seller **44**, to use different policies with different trading partners. For example, an seller **44** may use a VMI policy with one buyer **48** (for example, a retailer) and may use a non-VMI policy with another buyer **48**. As with the use of different policies in different tiers of the value chain, this disconnect in the policies in use between the seller **44** and the two buyers **48** also introduces inefficiencies.

In addition to problems associated with the use of different replenishment policies, problems also arise when attempting to obtain data from the various participants **30** as input to one or more planning or execution engines **50**, such as a replenishment planning engine. Such engines **50** can add significant value in the planning of activities and flows within the value chain; however, it has been problematic to gather all the data from the various participants **30** in a consistent fashion and to integrate the engines **50** with each participant **30**. Furthermore, in situations where planning or execution engines are run in a batch mode (often due to the lack of real-time data from participants **30**), participants **30** have not able to leverage the real-time feed-back of inventory changes occurring in the value chain. In short, most inventory management systems lack connectivity between participants **30** and do not provide adequate visibility into inventory information of the participants **30** in the value chain. This leads to increased production costs and reduced profits through the value chain due to problems such as unanticipated stock-outs, decreased production efficiency, higher investment in safety stock, increased obsolete and excess inventory, and inventory being in the wrong locations in the value chain.

The use of DIMS **20** reduces or eliminates these inefficiencies and allows each participant **30** in a value chain to simultaneously lower inventory, improve responsiveness, and lower costs. Additionally, participants **30** can collectively understand how relationships and interactions need to be modified in order to achieve or improve on desired objectives. DIMS **20** provides participants **30** with one logical distributed inventory "backplane" which has the capability to model engagement rules to enable interoperability of fulfillment/replenishment policies in different tiers of a value chain or co-existence of multiple fulfillment/replenishment policies for a given participant **30** in a value chain. This is accom-

plished by providing a framework to construct and maintain a virtual unification of the different systems against which business intelligence can be written.

Each participant **30** in a value chain typically generates exposed or exposable information relating to that participant's **30** physical elements within the value chain (such as warehouses, manufacturing facilities, stores, and delivery trucks). For example, this information may include information regarding which products are in which warehouse of a participant **30** or which orders map to which lots on a particular truck of a participant **30**. Each participant **30** also may create exposed or exposable business logic and operations at the element level of the supply chain (for example, purchase order acceptance and generation functions and order processing functions). However, this information is often not available throughout the value chain. DIMS **20** collects this information and business logic to provide a unified and normalized data model of the inventory information available at all participants **30** in the value chain that are coupled to and communicate with DIMS **20**. For example, the data model may identify which products are in which warehouses across the entire value chain or identify which orders map to which lots on which truck at any point in the entire value chain. Therefore, DIMS **20** may provide any participant **30** visibility, typically on a permissions basis, into inventory information of any other participant **30**. In addition, this unified and normalized information may be communicated to one or more engines **50**, and the output from the one or more engines **50** may be provided to any relevant participants **30** for planning and execution purposes.

FIG. **2** illustrates an example DIMS **20** in further detail. DIMS **20** includes several components or modules that perform various inventory management or related functions. DIMS **20** may be implemented as any appropriate combination of software and/or hardware operating in one or more locations. In one example embodiment, all of the modules associated with DIMS **20** are executed on one or more computers associated with a particular participant **30**. In another example embodiment, certain modules may be executed on one or more computers associated with a first participant **30**, other modules may be executed on one or more computers associated with a second participant **30**, and yet other modules may be executed on one or more computers associated with a third party (such as in an e-commerce marketplace). Furthermore, all of the modules may be executed on one or more computers associated with a third party. Any other appropriate location and distribution of the modules may also be used.

One module that the example DIMS **20** includes is an integration module **60**. Integration module **60** serves to integrate DIMS **20** with various information systems **32** of participants **30** and with one or more planning and execution engines **50**. This integration allows inventory data and other related information to be received and processed by DIMS **20** and, where appropriate, engines **50**. Integration module **60** allows information to be communicated between DIMS **20** and/or engines **50** to information system **32** or other systems associated with participants. Integration module **60** may handle connectivity to upload and download data in the context of the role-based or system-based workflows. The integration may be at a custom API level, as well as conforming to industry standards like EDI, RosettaNet™, and the like. Based on the characteristics of the various participants **30** that need to interact, integration module **60** may be configurable to interact on a transaction-by-transaction basis, a net change basis, using a complete batch refresh of data at specified time intervals, and/or on any other basis.

Integration module **60** may include appropriate components for communicating with and accessing data stored at or communicated from participant systems **32**. Systems **32** may include resource planning (ERP) systems **32***a*, legacy systems **32***b*, warehouse management systems (WMS) **32***c*, decision support systems (DSS) **32***d*, and/or any other appropriate type of systems operated by one or more participants **30** that are associated with the management of inventory and related information. Appropriate components of integration module **60** may receive or request inventory data from one or more systems **32** associated with one or more participants **30** and may then communicate this information to other modules of DIMS **20** and/or to engines **50** for processing. Therefore, integration module **60** may also include one or more components for communicating with and accessing data stored at or communicated from engines **50**. Engines **50** may include order management engines **50***a*, demand fulfillment engines **50***b*, supply chain planning engines **50***c*, factory planning engines **50***d*, replenishment planning engines **50***e*, and/or any other appropriate type of planning and/or execution engines that may be useful for the management of inventory and related information. As described above, these engines **50** may be associated with one or more participants **30** or with a third party (such as with an e-commerce marketplace).

Integration module **60** may also perform translation of data received from a system **32** and/or an engine **50** as appropriate for the destination to which the data is to be sent. For example, integration module **60** may translate the data format of inventory data received from an ERP system **32***a* so that it may be used by other modules of DIMS **20** and/or a replenishment planning system **50***e*. Planning data received from the replenishment planning system **50***e* may then be translated back to a format appropriate for the ERP system **32***a* and/or any other systems **32**. Integration module **60** may also or alternatively perform any other appropriate types of translation. Furthermore, integration module **60** may manage requests for data from systems **32** and/or engines **50** and responses to these requests. For example, integration module **60** may provide authentication and authorization of such requests. Integration module **60** may also serve to queue requests and responses, if appropriate, for communication to a suitable destination.

The example DIMS **20** also includes an administration module **62** that is used to configure and set-up the users of DIMS **20** which includes configuring the identity of the users of the system (participants **30** in a value chain) and their role-based permissibility as well as the frequency and types of messages that DIMS **20** is going to receive from participants **30** and other external sources. Two primary aspects of administration module **62** are a relationship modeling module **64** and an item modeling module **66**.

Relationship modeling module **64** is used to specify parameters associated with the inter- and intra-enterprise participants **30** in the value chain. The transactions and the data permissibility are governed by the rules configured using module **66**. Participants **30** that may be modeled include the various divisions of an enterprise and their locations (including location hierarchies) within the enterprise, as well as the customers and the suppliers that the enterprise interacts with. In essence this is the physical map that lays out the various participants **30** with which DIMS **20** is to interact, as well as their locations, roles, business rules, and/or any other appropriate parameters. Module **66** may also be used to implement agents, as described below.

Item modeling module **66** is used to specify the items relevant in the value chain, their descriptions, the locations at which the items are active, the item hierarchy, the unit of measure (UOM) of each item (and any hierarchies for the

UOMs), and/or any other suitable parameters relating to the items. Module **66** also plays a role in the aggregation of data, described below, in the sense that cross-referencing rules for the items may be specified using this module **66**. The ownership aspect of inventory is also specified using module **66**.

The example DIMS **20** also includes an inventory transactions module **68** that provides industry specific-configurable transaction mappings. The transactions may also have the capability of being solution-specific within an industry. All relevant transactions may be defined using module **68**. These transactions may then facilitate interaction with systems **32** and engines **50**. Module **68** also provides views of transactions as well as techniques for manually creating and editing transactions. A transaction history may also be maintained for audit trail purposes.

Furthermore, the example DIMS **20** includes an distributed inventory modeling module **70** and perpetual inventory modeling module **72**. Distributed inventory modeling module **70** provides a unified view of inventory information across the distributed systems **32** of participants **30** (which, as described above, may be intra-enterprise situation and/or inter-enterprise entities). Distributed inventory modeling module **70** provides a configurable and expandable inventory state map definition framework. This is useful since modeling a distributed environment typically requires interfaces with disparate types of systems **32**. Also, module **70** provides the ability to model computed and instance-specific states of inventory to support various industry solution workflows. Perpetual inventory modeling module **72** functions to act as a complete system supporting all the functions of a enterprise inventory management system. Perpetual inventory modeling module **72** functions to act as a complete system supporting all the functions of a enterprise inventory management system, such as transaction reconciliation, attribute level lot tracking, and tracing functions.

The example DIMS **20** also includes an inventory visibility module **74** that is integrated with inventory modeling modules **70** and **72** and inventory transaction module **68** to provide users with views into the distributed inventory environment. Module **74** also provides user-configurable reports that allow users with different roles to quickly access relevant data and make informed decisions. Based on information from modeling module **70**, a users can select a relevant portion of the inventory data regarding to view. Furthermore, the views may also provide information regarding the underlying transactions affecting the inventory in the value chain. Based on the permissibility framework established using relationship modeling module **64**, participants **30** may able to only view inventory which they have permission to view. Module **74** also provides the ability to define user-defined event triggers based on the distributed inventory information, thus enabling proactive management of inventory.

In addition, the example DIMS **20** includes a business analytics module **76** that provides the analytics services for both inventory transactions and for past, current and future inventory positions. For example, transaction audit trails may be used to view lot track and trace information. Furthermore, trending and profiling information may be provided using module **76**. For example, module **76** may be used to track the performance of new products as well as products that are being phased out, may be used to analyze liability and exposure in cases of consigned inventory, may be use to keep and track participant performance metrics, and/or may be used to perform any number of other suitable business analyses.

Although particular example modules have been described, it should be understood that DIMS **20** may include additional module, may not include some of the example modules, and may implement particular functions in a different manner than described above. The various modules of DIMS **20** collectively provide a common, distributed backplane between multiple participants **30** in different tiers of a value chain. This common backplane enables each participant **30** to acquire relevant information from other participants **30** related to inventory management and enables multiple participants **30** to provide data to a planning or execution engine **50** and receive the output of that engine **50** that is generated using the input of the multiple participants **30**. For example, an seller **44** is able to obtain information on planned shipments of components from suppliers **42** and information on planned purchases by buyers **48**, and thus is able to manage its inventory accordingly.

By enabling the common distributed backplane between the various participants **30**, a seller **44** (or any other participant **30**) is able to manage products being fulfilled and replenished using different policies through a single gateway. This provides the seller **44**, for example, with a consolidated and accurate status on the state of affairs of a particular product with respect to multiple buyers **48** even though different inventory management policies may be used with each buyer **48**. Service providers **42**, **46** can also be updated with respect to the planned movement of goods in the value chain. Therefore, all participants **30** potentially can have visibility into any appropriate information needed and can get this information from a single source, regardless of the inventory management program(s) in which the participant **30** is involved. This process creates efficiencies in the value chain by reducing or eliminating the need to plan against uncertainties created due to the lack of information related to inventory.

DIMS **20** also provides for different user roles (such as brand manager, fulfillment manager, and replenishment planner) that provide users with a customized view of inventory across various geographical and product hierarchies. For example, a brand manager might be only interested in getting visibility into safety stock levels across all locations and to be alerted to the possibility of a channel being starved or carrying excess. A replenishment planner, in addition to the information required by a brand manager, might also need visibility into in-transit inventory and an average weekly forecast. The use of DIMS **20** to model the value chain enables this custom visibility to different users. For example, geographic hierarchies can be modeled and thus a user of DIMS **20** can log on and request inventory information on a given product family for a given geographic region. In this case, inventory positions for that product family can be aggregated across all the warehouses in that region (and also from the stored connected to the warehouses, if this information is provided). The user would then be able to drill down to specific locations, and even get details by buyer and buyer location. DIMS **20** may also enable modeling of product hierarchies. In this case, for example, a brand manager can acquire visibility to inventory at a brand level and a replenishment planner can get visibility at a product stock keeping unit (SKU) level.

DIMS **20** also allows for easy definition of business rules throughout the value chain. Different users can easily monitor the value chain by receiving notifications of violations of the business rules. These rules can be based on minimum or maximum quantity inventory positions, computed minimum or maximum days of inventory based on projected sales-out and ending inventory, or based on numerous other criteria. Upon violation of a rule, DIMS may communicate a notification to a designated list of users. Furthermore, a violation of a rule may trigger a business process. As an example only, in scenarios like VMI, a violation of a minimum quantity rule may result in an alert being communicated to relevant partici-

pants **30** and/or a replenishment order being generated using an order management system. The generation of such a notification and the initiation of business processes may be performed using agents, as described in further detail below.

Furthermore, business logic may exist in DIMS **20** that serves as an intelligent mediator between systems **32** and engines **50**. For example, the intervening business logic may: evaluate incoming changes to determine if re-planning or aborting of an ongoing planning cycle is warranted, reconcile plan results with changes made in the model after a planning snapshot is taken, compare plan results with the current model state to compute net changes for communication to the chain elements, and/or direct messaging or model modifications to apply the model in accordance with the computed plans. The resulting combination of DIMS **20**, batch mode replenishment planning engine(s) **50***e*, and the logic described above provides a quasi-adaptive replenishment planning solution. Such a solution provides benefits such as flagging of exceptions in mismatches when different policies are used in different tiers in the value chain, use of a common inventory backplane to manage products under different policies with different participants **30**, an efficient way to evolve or change from one policy to another, and an efficient way to evolve from a batch-based fulfillment/replenishment planning paradigm to an adaptive fulfillment/replenishment planning paradigm.

FIG. **3** illustrates an example method for replenishment planning and execution using DIMS **20**. Using DIMS **20**, automated replenishment planning and execution in the multi-party inventory chain (including some or all participants **30** of a value chain) can be addressed by using data collected from the multiple participants **30** in a replenishment planning engine **50***e* and/or by using business logic associated with DIMS **20** to communicate the resulting inventory plans to the relevant participants **30** in the inventory chain.

The example method begins at step **100** where one or more participants in the value chain communicate with administration module **62** to specify set-up information for the value chain to be managed. The set-up information provides DIMS **20** with information regarding the make-up and characteristics of the value chain so that DIMS **20** may be used to manage the value chain. As an example, set-up information provided by a supplier **40** may include, but is not limited to: information regarding products, product components, or other items included in the value chain to be managed and information regarding the participants **30** in the value chain (such as the identity and location of the various participants **30**, the role of participants **30**, the manner in which particular participants **30** interact, and business rules associated with participants **30**). Participants **30** may communicate with administration module **62** using any appropriate communication techniques.

At step **102**, participants **30** provide or update previously provided inventory data and related data (such as demand, supply, and capacity data) using systems **32** and/or other appropriate data sources. As an example, a seller **44** may update inventory planning data to reflect an excess or shortage at supplier **40**. Data is provided to DIMS **20** through integration module **60** which is operable to communicate with systems **32** and perform any necessary data translation for use by DIMS **20**. Data may be communicated to DIMS **20** on a as generated, as requested, on scheduled basis, or in any other appropriate manner. Furthermore, this communication of data may be based on a "push" or "pull" method. For example, systems **32** may communicate or "push" data to DIMS **20** when relevant data is available or on a scheduled basis. Alternatively or in addition, DIMS **20** may request or "pull" relevant data from systems **32** as needed or on a scheduled basis.

DIMS **20** may store the data in any appropriate data storage location associated with or independent of one or more of the modules of DIMS **20**.

At step **104**, one or more modules of DIMS **20** perform internal processing of the data provided by participants **30**. For example, the inventory data may be communicated to inventory visibility module **74** so that it may be organized and integrated into views that may be communicated to users associated with participants **30**. As described above, the data may be organized so that it may be presented to a user having a particular role and having permission to view particular data. Furthermore, inventory visibility module **74** may monitor the data or information generated from the data and determine when an exception has occurred and/or when an alert should be communicated to particular participants **30**. Business analytics module **76** may also be used to analyze the inventory data for auditing, tracking, trend analysis, or for other types of data analysis. Processing may also occur in association with one or more agents associated with DIMS **20** that are responsible for performing particular business functions with respect to the inventory data.

Alternatively or in addition to performing step **104**, DIMS **20** may communicate inventory data to one or more engines **50** at step **106**. Engines **50** process this inventory data at step **108** using suitable techniques for the particular engine **50** and type of data. For example, inventory data from multiple participants **30** may be communicated to replenishment planning engine **50***e* for generation of a replenishment plan and/or schedule, inventory data may be communicated to an order management system **50***a* for generation of appropriate orders to replenish depleted inventory indicated by the inventory data, and/or any other suitable processing may be performed on the received inventory data by these or any other suitable engines **50**. Engines **50** may update an associated plan or schedule whenever inventory data or set-up information is changed by a participant **30** or on a periodic basis (for example, by requesting particular inventory data stored by DIMS **20**). A generated plan may also identify exceptions that have occurred due to a change in inventory data. One or more modules of DIMS **20** may also or alternatively analyze a plan or schedule and identify any exceptions (and generate alerts for particular participants **30**, if appropriate). At step **110**, engines **50** communicate the planning and/or scheduling information to DIMS **20**.

At step **112**, DIMS **20** communicates inventory information generated by DIMS **20** and/or engines **50** to appropriate participants **30** (such as to systems **32** of particular participants **30**) using integration module **60**. The inventory information may be communicated based on a request by a participant **30** (such as by a system **32** of a participant **30**), based on the generation of data by an engine **50** and/or a module of DIMS **20** that is relevant to a participant **30**, based on an alert generated for a particular participants, based on a pre-planned schedule, and/or for any other appropriate reasons. Therefore, this communication of information may be on a "push" or a "pull" basis. For example, a seller **44** may request a view of inventory of a particular item at one or more locations and such a view may be communicated to the user. As another example, a module of DIMS **20**, such as inventory visibility module **74**, may evaluate incoming inventory data from a supplier **40**, recognize that the inventory level of a particular item has dropped below a pre-defined level, and communicate an alert to the supplier **40** and an associated seller **44**. As yet another example, an engine **50** may identify an exception in a plan or schedule generated from inventory data received from DIMS **20** and DIMS **20** may communicate the existence of such an exception to appropriate participants **30**. In general,

DIMS **20** serves as a collection point for inventory information for multiple participants **30** in a value chain and DIMS **20** may serve to communicate the "raw" inventory data of one participant **30** directly to other participants **30**, process the data internally and communicate the results of this processing to one or more participants **30**, and/or communicate inventory data to appropriate engines **50** for processing and communicate the results of this processing to appropriate participants **30**. At step **114**, participants **30** receive information communicated from DIMS **20** and take appropriate action.

Although the steps of the example method described above, as well as those described below, are illustrated as occurring in a particular order, it should be understood that the steps may occur in any appropriate order and different steps may occur simultaneously for different instances of inventory data or for other suitable reasons. Furthermore, one or more of the steps may not be performed and/or additional steps may be include as appropriate for the operation of DIMS **20**.

As mentioned above, intelligent fulfillment agents may be used by DIMS **20** to conduct various tasks. These fulfillment agents facilitate the triangulation of relevant participants **30** in the context of the various fulfillment and replenishment workflows that occur between the participants **30**. Unique business relationships and rules are captured in configurable fulfillment engagement rules that leverage the inventory visibility provided by DIMS **20**. The triangulation takes place at two levels: the participant relationship level and the data/information level. In the former, the participants **30** involved are modeled (for example, a buyer **48**, a seller **44**, and fulfillment/execution service provider **46**). The data/information level contains the data and information includes the inventory and other related information provided by DIMS **20**. The fulfillment agents enable a seamless flow of information between these two levels thus provide an event-based replenishment transaction management with rich decision support. The agents are also intelligent in that they can adapt to execution level exceptions changes that occur, such as at a planning or strategic level in the value chain. For example, if changes to the network set-up occur or customer demand increases occur, these agents can recognize such changes and use the information about the inventory positions in the value chain to react as necessary with appropriate workflows.

The flexible framework will allow participants **30** to adapt to changing market needs and business rules. Two fundamental enablers of this framework are visibility and connectivity. The visibility provides a view of the inventory both at a discrete level (for example, by SKU, location, or owner) as well as an aggregated view across a distributed environment (for example, a single seller **44** with multiple sites or multiple sellers **44**). The transaction management capability of DIMS **20** will let the users track inventory as it changes ownership or is being co-managed by the various entities involved in the fulfillment and replenishment workflows.

Different fulfillment agents may be used for different types of fulfillment and replenishment workflows. For example, one agent may be used for a JIT workflow and another may used for a VMI workflow. Although many other types of agents may be implemented through DIMS **20** to enable any suitable fulfillment and replenishment workflows, for purposes of example a JIT fulfillment agent will be described below in further detail.

There are four main aspects of a typical JIT implementation. The first such aspect is a contract(s) between the relevant participants **30**. For example, in a JIT environment, a supplier **40** delivers raw materials and other purchased items as they are needed. A blanket purchase order or other suitable form of

basic agreement typically covers the terms and conditions for this procurement. Generally, the relevant participants agree on acceptance criteria before they embark on the JIT program. A set of key performance indicators (KPIs) are often selected to measure the effectiveness of the program and to set criteria for the participants **30** to follow. For example, KPIs may address the service level of the suppliers **40**, the inventory turnover, the fill rate, and/or any other suitable measures.

The next aspect of JIT is planning. JIT typically requires a flow of material in the exact quantity required at an exact time. Regardless of the specific method used to achieve this exact material flow, there must be advance planning to ensure that material is available when needed. A technique such as material requirement planning (MRP) is generally used to accomplish this task. In addition to predicting the future material requirements, just-in-time replenishment quantities and minimum/maximum levels need to be determined. Furthermore, early warning visibility and assistance in decision-making related to unexpected events is also needed.

A JIT implementation also involves electronic data interchange (EDI) message transmission and translation. In many value chains, value chain management extends to the sharing of planning, operational, tactical, and strategic data. This includes sharing short-term scheduling data, medium-term materials requirements planning and scheduling, and longer-term forecasts. The traditional technique for performing JIT communications has been the use of EDI over value added networks (VANs). The disadvantages of EDI-VAN are that the use of VANs is expensive, there are many trading partner-specific variations on EDI standards (and thus communication problems are created), and implementing EDI is a costly and technologically challenging process.

Configuration is also an aspect of JIT. Depending on the industry, there are many configurations possible. Some sellers **44** may want to own inventory and typically the suppliers **40** will deliver the material directly to a seller's location when needed. Another alternative may be that sellers **44** establish their own warehouses near a seller's facility and material is delivered in a just-in-time manner when required by a seller **44**. Some sellers **44** may opt to have a hub (typically a warehouse managed by third party service provider **42**) and suppliers **40** deliver the products directly to hub so that material can pulled on a just-in-time basis from the hub. In this case, either the supplier **40** or the third party service provider **42** owns the inventory depending on the business scenario.

JIT replenishments are normally focused on short-term order fulfillment. The following are example steps of a typical JIT fulfillment process. The seller **44** (such as a manufacturer) sends forecast information and a planning schedule to its suppliers **40** on a periodic basis (for example, weekly). Suppliers **40** use this information for their internal planning. These forecasts typically span a forward horizon of eight to twelve weeks. Seller **44** also sends a shipping schedule to suppliers **40** on a daily or weekly basis. Depending on the business practice, suppliers may or may not be required to provide commits against the shipping schedule. The shipping schedule in a traditional MRP is a push communication. It basically tells the supplier **40** what items the seller **44** wants the supplier **40** to ship. The seller **44** expects to use these items and it wants the supplier **40** to deliver the items whether they end up using the items or not.

On a daily basis, seller **44** monitors the inventory levels and demand signals and sends EDI change notifications to first tier suppliers **40** to adjust the shipping schedules. First tier suppliers **40** examine the contents of the schedules and determine whether or not they can meet the requested quantity. Each first tier supplier **40** then sends an acknowledgment to

seller **44**. First tier suppliers **40** also input the shipping schedule information into their planning systems to use the information as the basis for their forecast demand. However, daily shipping schedules constantly vary from forecast requirements, so safety stocks (inventory buffers) are added into the calculations and net requirements for raw materials and subassemblies or components are calculated. These planned net requirements for procured inventory are then communicated weekly to second tier suppliers **40** (suppliers to the first tier suppliers), sometimes via EDI or more commonly via fax. However, as a result of daily changes in OEM actual demand, first tier suppliers **40** frequently issue daily changes to their planned requirements from their suppliers **40** (the second tier suppliers **40**). This same process is repeated between second tier suppliers **40** and third tier suppliers **40**, and so on. However, EDI is rarely the communication medium between these lower tier suppliers. More typically, the communication medium is fax, phone, or mail. The net result is considerable delay in communicating requirements to lower levels in the supply chain. As a result, suppliers **40** to the first tier suppliers **40** try to respond to planned requirements several weeks in advance and compensate by building up safety stock levels to accommodate the eventual wide variability in demand.

The first tier suppliers **40** each generate an advanced ship notice once a truck going to the seller **44** is loaded. This transaction notifies the seller of the actual contents of the truck and can be helpful for scheduling the receiving dock and identifying cross-docking opportunities. Similarly, shipping notices also are sent between suppliers **40** at the different tiers. Again, the communication mode may be EDI, but is more commonly fax, phone or email in these cases. The truck carrying the product typically arrives at the seller's **44** dock at its fixed appointment time and is promptly unloaded so that the product can be available to the line. The replenishment lead-time usually is quite short and consists of mainly the in-transit time, which can range from a few hours to few days.

Although the JIT process described above may have advantages over other types of processes, there are also many challenges associated with this traditional JIT fulfillment process. Such challenges include demand variation challenges. Customers (which may include sellers **44** and suppliers **40**) generally push variation in demand down to their suppliers **40**. This is slow, cumbersome, and overly expensive because forecast data is batched and depends on heavy administrative intervention even where EDI systems are in use. This process inherently builds excess inventory to accommodate wide variations in daily demand. An insidious side effect to the volatility in daily demand is that production schedules are difficult to plan and maintain. This creates inefficiency and causes excessive overtime to be worked. Although a seller **44** provides the forecast data to its suppliers **40**, the shipping schedule (JIT material requirements) almost always differs from the forecast because of the demand variation.

There are also communication challenges associated with the traditional JIT process. Although a seller **44** and a first tier supplier **40** may have good relationships and may be communicating through well-established EDI communication programs, this EDI communication gets expensive as more and more participants **30** join the JIT program. In many instances this leads to custom data mapping and maintaining these mappings can get very expensive. On the other hand, the level of sophistication decreases as you move down the supply chain and the downstream suppliers **40** rely mostly on fax, phone and e-mail messages for communication with upstream suppliers **40**. In order to compensate for the inefficiencies with such communication techniques, more and more inventory is added to support upstream demand. Furthermore, due to improper communication set-up, delivery problems from lower tier suppliers **40** are often not communicated to upper tier suppliers **40**, or such problems result in fragmented, duplicated, or delayed communications. This affects the supplier reliability and creates a lack of trust throughout the value chain.

The traditional JIT process also has visibility challenges. Since traditional JIT fulfillment is an independent process, there is not visibility into any of a seller's planning or transactional data. It also does not contribute to increasing the visibility of the seller's planning processes. Lack of visibility into multi-tier participant inventory (for example, a fourth tier supplier's lack of visibility into the inventory of a first tier supplier **40** or a seller **44**) poses a challenge to suppliers **40** since they have to deal with their customer's near term demand changes and end up building more "just-in-case" inventory to satisfy changing demand needs. Changes in order sequence, inventory buildups, delayed response from lower tier suppliers as well as transport delays could be avoided by providing global visibility to all appropriate participants **30**.

Because of the above challenges, first tier suppliers **40** spend too much time and money managing and directing their suppliers **40**. This approach is labor intensive, results in unreliable part deliveries, and generates high inventory levels in the supply chain. It also results in excessive expediting and high administrative costs. Furthermore, last minute demand and schedule changes result in high premium freights for the suppliers **40**. Suppliers **40** also have to consume the high inventory carrying costs that result from inventory they build to safeguard themselves from unpredictable variation in daily demand changes. All of this adds significant cost to the value chain.

The use of a DIMS fulfillment agent for JIT fulfillment can reduce or eliminate the above problems with the traditional JIT fulfillment process. JIT fulfillment agents and other fulfillment agents associated with DIMS **20** may be implemented as software associated with DIMS **20** or may be associated with a participant **30** an be programmed to use information provided through DIMS **20** to accomplish one or more specific business functions that are the responsibility of each agent. As described above, DIMS **20** provides visibility into a buyer's demand signals and planning forecasts for all relevant participants **30** in the value chain. JIT fulfillment agents associated with DIMS **20**, as well as the use of other types of agents provided through DIMS **20**, may serve to automate the process of monitoring relevant information from participants **30**, provide exception monitoring and generation of automatic alerts for participants **30** as and when particular events occur (such as alerts generated when inventory levels are violated), and initiate appropriate business processes in response to this information and events.

Specifically, JIT Fulfillment agents can be used to employ different business models ranging from single-tier model to multi-tier models. In both cases, participants **30** may have one-to-one (a single seller **44** interacting with a single supplier **40**), one-to-many (a single seller **44** interacting with multiple suppliers **40** and vice versa) and many-to-many (multiple sellers **44** interacting with multiple suppliers **40**) relationships. The basic concept of JIT fulfillment remains the same whether it is a single tier or multi-tier value chain or whether one-to-one or one-to-many relationships are modeled. Although the concepts described herein primarily relate to the relationships between a seller **44** and a supplier **40**, these concepts are applicable between any two participants **30** in a value chain.

As described above, DIMS **20** provides an integration service that communicates with systems **32** to get the relevant inventory and related business data as and when required. All relevant data may be stored at a single location (for example, at a seller location **44**) or may be distributed at multiple locations (but integrated) and is made available to agents to automate businesses processes, such as a JIT process. Using such agents, a company can interconnect all of its trading partners and keep everyone synchronized with each other's current plans while tracking actual activity in real-time across companies. This increases the velocity in the value chain and results in increased efficiency, increased planning accuracy, improved ordering and inventory control, reduced inventories (and thus costs) throughout the supply chain.

FIG. **4** illustrates an example JIT fulfillment workflow that may be implemented and automated using JIT fulfillment agents associated with DIMS **20**. The method begins at step **200** where a JIT agent associated with or in communication with DIMS **20** initiates and triggers a request from DIMS **20** to a seller **44** (such as a manufacturer) for a planning schedule for long-term planning (such as an EDI **830** message) and for a product inventory report (such as an EDI **852** message) at a predefined time interval (for example, daily or weekly). At step **202**, the seller's inventory management system communicates the requested planning schedule and product inventory report to DIMS **20**. This information may be communicated to VMI **20** in response to the request or at an appropriate user-defined time interval. Integration module **60** allows the receipt of this information from the seller's inventory management system **32** and may convert the information (for example, in an EDI message or messages) to an appropriate format for processing. The JIT agent may direct this translation based on rules for data conversion for the particular JIT program.

At step **204**, DIMS **20** persists the scheduling data and updates the seller's stock levels for all SKUs based on the product activity data. At the direction of the JIT agent, DIMS **20** communicates the seller's forecast update based on the schedule to the suppliers' inventory management system **32** at step **206** (the communication may be translated if appropriate for each supplier **40**). Alternatively, an agent may exist at the supplier **40** and have rules for data and format conversion for that particular supplier **40**. At step **208**, the relevant suppliers **40** use this data to plan their forecast and, in turn, communicate their forecast to downstream suppliers **40** using the JIT agent and DIMS **20**, as described above.

At step **210**, the JIT agent monitors the seller's stock levels communicated to DIMS **20** and sends an alert to the supplier at step **212** based on a JIT reorder strategy implemented as rules associated with the agent. Such an alert may communicated to a system **32** or directly to a user via e-mail, page, fax, or using any other appropriate communication technique. In addition or alternatively, the agent can create a shipping or delivery schedule (such as an EDI **862** message) and communicate this schedule at step **214** to relevant suppliers **40** when the inventory level reaches a pre-set just-in-time reordering point. Such a schedule may be communicated to the seller **44** for confirmation and seller **44** may modify it and upload the modified schedule to DIMS **20** at step **216**. Furthermore, at step **218** the agent may communicate relevant information to order management system **50***a* for the creation of a purchase order for the items that are needed to increase the stock level. At step **220**, order management system **50***a* communicates one or more appropriate orders to DIMS **20**, and agent directs the communication of these orders to appropriate suppliers **40** at step **222**.

At step **224**, the relevant suppliers **40** communicate an acknowledgement of the alert, shipping schedule, and/or purchase orders. If the supplier cannot meet the shipping schedule, an exception workflow may be triggered. At step **226**, the suppliers **40** perform transportation planning for the orders and communicate an order status to DIMS **20**. The JIT agent directs the communication of the order status to seller **44** at step **228** to update the seller's order management system. Similarly subsequent order statuses like advanced shipment notices from suppliers **40** may be updated for an order in DIMS **20** and relayed to the seller **44**. At step **230**, the JIT agent also requests and receives the order status from relevant service providers **42** during transportation to update both the seller **44** and supplier **40**. The JIT agent tracks the order until the status of the order is considered closed by both the participants **30**, and the example method ends. All of the actions of the JIT agent described above may be the result of rules associated with the agent that dictate the action the agent takes when a particular information is received from a participant **30** (such as through a system **32**), from an engine **50**, and/or from any other appropriate source.

Although a JIT fulfillment agent is described in the example method above, numerous other types of agents may be used in conjunction with DIMS **20** to implement rules to execute a particular business function. For example, agents can be programmed to execute other replenishment programs, such as VMI or SMI, in a manner similar to that described above. Furthermore, agents can be programmed to evaluate data received from participants **30** and/or engines **50** and to generate alerts based on rules that are used to identify issues with the data (for example, maximum and/or minimum levels of a particular data measure, violation of timing-based tolerances, or mismatched in ordered and delivered items). Agents may also generate alerts based on the absence or presence of a transaction (for example, the non-receipt of an expected EDI message). Agents may also be used to perform and/or initiate a business process, such as the creation and communication of a purchase order using an order management system based on an evaluation of inventory data from one or more participants **30**. As described above, agents may also automate a series of tasks, such as automating the numerous tasks involved with a replenishment program (such as the JIT program described above). Agents may further be used to automate any other appropriate tasks associated with inventory management and thus may be used to efficiently leverage the inventory visibility provided by DIMS **20**.

In addition to the advantages described above, the connectivity and visibility provided by DIMS **20** may also be used to perform distributed order fulfillment. Order fulfillment is the process of managing a customer order through its entire life cycle. Broadly classified, the life cycle of the order spans the processes of capture and creation, verification and approval, quoting, sourcing, planning and scheduling, execution, and returns management. The specifics within these order processes often vary to a high degree between industry segments and also between different sellers in the same industry segment. However, these different workflows have some basic similarities that can be traced to the business models followed by these sellers. These business models and their influence on existing order fulfillment processes are described below as context to the description of the novel distributed order fulfillment process made available using DIMS **20**.

One aspect of order fulfillment is intra-enterprise fulfillment between the different portions of an enterprise. Intra-enterprise order fulfillment may be a challenge for sellers **44** having multiple divisions each having a separate order fulfillment process. These divisions typically may have dispar-

ate systems managing their order fulfillment life cycle and there has often been little visibility and coordination between these divisions. Similarly, a seller **44** may have multiple stocking locations for a product and ship from any of these locations depending on the buyer **48** and destination of an order. However, once a sourcing location has been identified and the order information has been sent to this location, the seller **44** may loose visibility to the order until it is actually delivered to the buyer **48**. Having visibility to an order during order planning and execution steps and managing any exception thus becomes an expensive and time-consuming process for the seller **44**.

For example, many large sellers **44** typically have multiple divisions that are managed and operated independently. These divisions may be organized by the product lines or by regions or by industry segments. Many of these divisions have different operating practices and business processes in place. They may also have disparate seller applications managing these business processes. As a result, it is difficult to get an aggregated view of data associated with the seller **44** across all these divisions. Because of the lack of an integrated environment, such sellers **44** often direct a buyer **48** to place an order with the division concerned and buyer **48** ends up sending multiple orders to different divisions of the same seller **44**. This results in an increased ordering cost for the buyer **48** and increased order maintenance cost for seller **44**. Some sellers **44** do use a single Internet store-front that allow a buyer **48** to place an order and then internally direct the order to the right division, but such a system still lacks the visibility at the back end of the supply chain where the actual order fulfillment takes place. This makes the order coordination across divisions a difficult task for a seller **44**.

Furthermore, it is typical for a seller **44** to have multiple stocking locations from where a product can be shipped to a customer. The stocking locations may be distribution centers in different regions and/or warehouses near the seller's manufacturing plants. Depending on the number of these locations and the products stored each location, the sourcing decision for a customer order can become complex to manage effectively. Most of these sourcing decisions are static in nature, such as a distribution center in a region supplying to the customers in that region to minimize the transportation cost. These static rules often lead to situations where a customer order is delayed in one distribution center because of product shortage even though a distribution center in a different region may have excess of the product.

Problems also exist with order fulfillment in an multi-enterprise ordering environment. In such an environment, buyers **48** depend on a multitude of sellers **44** for various products (or sellers **44** depend on a multitude of suppliers **40**). Because these sellers **44** are different entities, the buyers **48** work with these sellers **44** independently to manage the order fulfillment process. For example, a buyer **48** would typically have a contract in place with a number of sellers **44** of different products. During an order fulfillment process, the buyer **48** would place purchase orders separately to these sellers **44** for the specific products that they supply. The buyer **48** would then follow-up independently with each of these sellers **44** during the order life cycle. This leads to increased ordering cost and makes the order coordination between sellers **44** extremely difficult.

Most of the order fulfillment challenges for participants **30** in a value chain result from the lack of visibility into each other's supply chain and lack of information sharing between participants **30**. Buyers **48** often do not get the right promise from their sellers **44** on their orders and thus do not get the product at the right time. This forces them to maintain a higher level of inventory for the critical products or components, which leads to higher carrying cost. Similarly a buyer **48** often cannot place a consolidated order to multiple sellers **44** and be able to track the order through its whole life cycle. This results in higher ordering cost and order management cost.

Many of sellers' fulfillment-related problems stem for an inaccurate demand information from a buyer **48**. Even when the demand planning is within acceptable limits, the sellers **44** still incur major fulfillment-related problems because their material, product, and/or capacity profiles are unreliable. The most common causes of this include internal production or procurement problems, in addition to poor communications with the buyers **48** (often due the high connectivity costs). These problems contribute to excessive fulfillment cycle time, expediting costs, poor margins and service levels and, ultimately, lost business.

DIMS **20** may be used to provide a distributed order fulfillment service that provides participants **30** with the infrastructure and a configurable set of workflows to manage the entire life cycle of a distributed order (for example, a multi-seller or multi-divisional order). The distributed order fulfillment process implemented using DIMS **20** may be used to manage the entire life cycle of an order starting with order capture and promise and ending with a proof of delivery and subsequent financial settlement of the order. To provide these functions, appropriate order management and fulfillment modules may be added to or associated with DIMS **20** in particular cases or such functions may be implemented, in part, through appropriate engines **50** in communication with DIMS **20**. This distributed order fulfillment process provides fast and accurate order promising, real-time order status reporting for a seller **44** and for its buyers **48**, dynamic sourcing decision-making for optimal order fulfillment, and efficient exception management and error handling.

FIGS. **5**A and **5**B illustrate an example distributed order fulfillment method. As described above, order fulfillment workflows may vary greatly between the industry segments and also between different sellers **44**. Therefore, an example generic distributed order fulfillment process is described below. This example process may be divided into two segments: order fulfillment and order execution. Order fulfillment includes the processes of capture and creation, verification and approval, quoting, sourcing, and scheduling of an order. Order execution includes the processes of warehouse planning and execution and transportation execution and management.

The method begins at step **300** where a buyer **48** creates an order for an item, such as a product. Although a buyer **48** is described as purchasing items from sellers **44**, it should be understood that the process applies equally to sellers **44** purchasing items from suppliers **40**, suppliers **40** purchasing items from other suppliers, and any other appropriate transactions between participants **30** in a value chain. Some of the parameters included in the created order may be the requested item(s), the corresponding due date and quantity of each item, and the desired sellers **44** for an item (for example, the buyer **48** may have existing contracts with particular sellers **44**). Buyer **48** may alternatively specify a list of sellers **44** who are eligible to supply an item in the order.

Once the order is created, buyer **48** communicates the order to DIMS **20** at step **302** for either a quote (using a request for quote) or a promise (using a request for promise). For example, if buyer **48** is EDI-enabled, buyer **48** may send an EDI **840** message to DIMS **20** for a quote or an EDI **850** message to place a confirm order. For the purposes of this description, it is assumed that buyer **48** first communicates a

request for quote. At step **304**, DIMS **20** may send a functional acknowledgement (such as an EDI **997** message) to buyer **48** upon receipt of the request. At step **306**, DIMS **20** determines, using the information about participants modeled in DIMS **20**, which sellers **44** should receive the request based on the items requested in the order and, if applicable, the set of potential suppliers requested by buyer **48**. Depending on the number of items included in an order and the sellers **44** requested by buyer **48**, each seller **44** might receive one or more line items for a quote. This function, as well as the other functions of DIMS **20** described below, may be performed using one or more of the modules described above and/or using one or more additional modules (such as a fulfillment server) added specifically for distributed order fulfillment functions. Again, the various modules of DIMS **20** may be co-located at one location (for example, at a location associated with a participant **30** or a third party) or they may be located at multiple locations (and potentially replicated at multiple locations). For example, relevant sellers **40** may be identified using data from relationship modeling module **64**. If the terms of the buyer's contracts with one or more sellers **44** are communicated to and stored by DIMS **20**, DIMS **20** may access these contractual terms to determine appropriate sellers **48** for the buyer's requested items and the corresponding prices for each item, if specified in the terms.

At step **308**, DIMS **20** communicates the buyer's request to the appropriate sellers **44** (if a seller **44** is EDI enabled, DIMS **20** may communicate an EDI **840** message to the seller **44**). At step **310**, the selected sellers **44** receive the request for quote and generate a response to the quote. The generation of the response is an intra-seller task and the mechanism for generating such a quote may range from a completely manual process to the use of a sophisticated order-quoting engine, such as a demand fulfillment engine.

At step **312**, DIMS **20** may receive the responses from the sellers **44** (possibly over a window of time specified by buyer **48** during which responses will be accepted). DIMS **20** consolidates these responses into a single response to the original request at step **314**. The business logic of consolidation of the individual responses is configurable by buyer **48**. For example, one buyer may want to see all the responses from all the sellers **44** and then select the best response, whereas another buyer may prefer that DIMS **20** determine, based on pre-configured business rules, the best response to the request. These business rules may be implemented in an agent or using any appropriate modules of DIMS **20**. At step **316**, DIMS **20** communicates the consolidated response to buyer **48**. If buyer **48** is EDI-enabled, an EDI **843** message may be sent to buyer **48**.

Upon receiving the response, buyer decides at step **318** whether to reject the quote or place a confirm order against the quote. If buyer **48** decides to place a confirm order, buyer **48** communicates a purchase order to DIMS **20** at step **320** (if buyer **48** is EDI-enabled, buyer **48** may communicate an EDI **850** message for order confirmation). It is assumed that the purchase order creation is an intra-enterprise activity for buyer **48**. The mechanism used by buyer **48** to create a purchase may include a manual process, an ERP system, an order management system, or any other appropriate mechanism. At step **322**, DIMS **20** receives the purchase order and may validate the order according to business rules established between buyer **48** and a seller or seller **44**. At step **324**, DIMS communicates an acknowledgement of the receipt of the purchase order (such as an EDI **997** message).

Once the order is received by DIMS **20**, it is persisted in DIMS **20** or an associated engine **50** (such as an order management engine **50***a*) and is given an appropriate order status at step **326**. At step **328**, if the order is based on a previous quote, DIMS **20** assigns a promise to the order based on both earlier responses by the sellers **44** and/or the buyer's selection of a quote and the order status is adjusted appropriately to indicate the promised state. In addition or alternatively, DIMS **20** may provide a real-time promise to buyer **48** on behalf of a seller **44** based on inventory information provided to DIMS **20** by the seller **44**. Therefore, DIMS **20** provides buyers **48** with a selective view to the sellers' inventory to help plan their own orders. If a seller **44** is not used for final confirmed purchase order, DIMS **20** may notify the seller **44** with an appropriate message at step **330**. At step **332**, DIMS **20** communicates a message to the buyer (such as an EDI **855** message) for purchase order acknowledgement.

At step **334**, DIMS **20** determines appropriate destinations to which the order is to be sent and communicates the order to the appropriate destinations associated with sellers **44** at step **336**. Each seller **44** receives a purchase order that only consists of the line items concerning that seller **44**. If the supplier is EDI-enabled, DIMS **30** may communicate an EDI **850** message to seller **44**. As described above, a seller **44** may have multiple independent divisions or multiple warehouses supplying a specific item. In addition, the seller **44** may have multiple suppliers **40** that can either supply the item to the seller **44** or drop-ship the item to the buyer **48** for the seller **44** if needed. In scenarios like this, the seller **44** has previously needed to search through all of the divisions and/or warehouses to find out the item availability and the sourcing location. The sourcing logic can become extremely complex and can become unmanageable for a large seller **44**. In addition, once an item it is sourced to a division or warehouse, the seller **44** may loose visibility to the order at the seller level because of the lack of visibility to the divisional operation. However, DIMS **20** provides sellers **44** with a scalable and configurable framework to implement these complex sourcing rules and automate the sourcing process.

For example, assume that a buyer **48** requests that an item be shipped to its New Jersey distribution center from a seller **44** that stocks the item in distribution centers based in Los Angeles and Chicago. If minimizing the transportation costs is one of the sourcing criteria, then the business rules can be configured in DIMS **20** to source the item from Chicago instead of Los Angeles (assuming it will cost less to ship from Chicago). As another example, DIMS **20** may source the order from a the warehouse of a seller **44** having the most unpromised inventory of the ordered item (according to the inventory data provided to DIMS **20**). DIMS **20** provides sellers **44** a scalable framework to conveniently model the various sourcing rules that may be used. Once the sourcing logic is configured in DIMS **20** for a seller **44**, then the purchase order from buyer **48** can be communicated directly to the particular division or warehouse selected at step **336**.

At step **338**, each seller **44** (as described above, the term seller as used here may refer to a particular division of a seller **44** or other destination associated with a seller **44**, and each destination may be modeled in DIMS **20** as a participant **30**) receives the appropriate order and may compare the original promise made in the response to the request for quote (made by a seller **44** and/or made by DIMS **20** on behalf of seller **44**) to the latest promise based on the latest supply position. If the promise is different, then the seller **44** may initiate a change promise workflow at step **342** and notify the buyer **48** via DIMS **20** (in cases when the purchase order is not a result of a quote process, seller **44** sends the promise information to DIMS **20**). If the seller **44** is EDI-enabled, it may send an EDI **855** message (purchase order acknowledgement) to DIMS **20** to update the earlier promise or send a new promise. At step

344, DIMS **20** updates the promise for the seller **44**, if needed, and notifies buyer **48** regarding any change (for example, by communicating an EDI **855** message to buyer **48**). This completes the order fulfillment process for the order. It should be noted that while a seller **44** is responsible for only one or more line items of the buyer's purchase order, DIMS **20** keeps track of the buyer's complete purchase order and provides buyer **48** with real-time order status updates, as described below.

FIGS. **6**A and **6**B illustrate an example distributed order execution method. Order execution includes the processes of warehouse planning and execution and transportation execution and management. The order execution process typically involves tasks performed by the seller **44** and associated third party service providers **46** to deliver the order to the buyer destination (although, as described above, the process applies equally to sellers **44** purchasing items from suppliers **40** and using service providers **42**, suppliers **40** purchasing items from other suppliers **40** and using service providers if appropriate, and any other suitable transactions between participants **30** in a value chain). Although these tasks are mainly intra-enterprise to the seller **44**, the status of the order is keep updated and received by buyer **48** and other appropriate participants **30** using DIMS **20**. Furthermore, the various communications between the participants **30** described below may be communicated directly between the participants **30** and/or via DIMS **20**.

An order is often shipped by a seller **44** in shipments or loads. Grouping of orders into a load or shipment allows the seller **44** to minimize its transportation cost. Transportation planning is the process where the seller **44** groups the orders into optimized shipments or loads based on a set of criteria such as destination, source, routes, transportation cost, special handling instruction, and any other appropriate criteria. The transportation planning process may also take into account any merge-in-transit and/or value added service constraints on the order.

The example method begins at step **400** where the seller **44** carries out appropriate transportation planning for an order by grouping the order into loads or shipments. The outcome of such a planning process is a set of loads that consist of one or more orders with one or more items. Once the loads are created, the loads are tendered to one or more carriers **46** for transportation at step **402**. Load tendering is the process where carriers **46** are contacted with the load information (for example, source, destination, item, and pick-up/delivery window). The messages associated with transportation planning, as well as all other messages associated with order execution, may be communicated to the appropriate entity using DIMS **20** (and the message may be communicated directly to the entity, if appropriate). At step **404**, DIMS **20** receives the load tendering message and updates the status of the associated order (for example, the order status may change to "load tendered" status).

At step **406**, DIMS **20** communicates the load tendering message to appropriate carriers **46** and carriers **46** respond to the request. Seller **44** receives the responses (for example, using DIMS **20**) and selects a carrier **46** at step **408**. Seller **44** may have a predefined contract with a specific carrier **46** for a set of routes or a region. If seller **44** and a carrier **46** are connected via EDI, the seller **44** may communicate an EDI **204** to the carrier **46** and carrier **46** may send an acceptance using an EDI **990** message (these messages may be sent directly or via DIMS **20**). At step **410**, DIMS **20** receives these messages and updates the status of the order based on the messages (for example, the order status may change "tender accepted" status).

Once a tender has been accepted by a carrier **46**, the loads are released at step **412** to an appropriate warehouse for warehouse order planning. The warehouse is also notified about the appointment time when the carrier's trailer should arrive at the warehouse to pick up the load. Seller **44** may send an EDI **940** message to a warehouse as the order release signal. DIMS **20** receives the order release communication at step **414** and updates the order status appropriately. Once a warehouse receives a release notice, the items related to the orders are identified, picked and packed at step **416**. This operation is usually a batch process and is done a few times during a typical day depending on the order volume. Any discrepancy that would cause an order to be shipped late is identified at step **418** and a message indicating any discrepancy is communicated to DIMS **20** at step **420**. At step **422**, DIMS **20** updates the order status appropriately and notifies the buyer about the exception. The buyer may then initiate a change order workflow, if desired.

Once the shipments are created, they are moved to the specified area or dock where the carrier is to load the order. At step **424**, the warehouse communicates an advanced ship notice (ASN) to DIMS **20** (for example, an EDI **856** message may be sent as an ASN). A typical ASN includes the order number, the items being shipped, the delivery date, the delivery window, and any other appropriate parameters. At step **426**, DIMS **20** receives the ASN and updates the order status. DIMS **20** communicates the update order status to buyer **48** at step **428**.

If a carrier **46** picks up a loads from the warehouse dock at the specified time, a confirmation message (such as an EDI **214** message) is communicated to DIMS **20** at step **430**. DIMS **20** receives this message and updates the order status appropriately at step **432**. If the carrier does not show up at the appointed time, the orders need to be re-planned for a future date and time. In such a case, an appropriate message is communicated at step **434** from the warehouse to DIMS **20** to identify this exception. DIMS **20** notifies the buyer of this exception through an exception management workflow at step **436**.

Once carrier **46** has picked up a load, carrier **46** may send a daily or other periodic en route shipment status updates to DIMS **20** at step **438**. At step **440**, DIMS **20** receives such updates and updates the status of the order. At step **442**, DIMS **20** may communicate the updated order status to buyer **48** as updated, on a request basis, or at any other appropriate times (in some cases, DIMS **20** may only communicate an update if an exception has occurred). Once the shipment is delivered to the buyer destination, the carrier receives a proof of delivery (POD) and communicates the POD (for example, and EDI **214** message) to DIMS **20** at step **444** to indicate the completion of the delivery process. An EDI **214** message may be sent for a POD confirmation. DIMS **20** receives the POD and updates the order status to indicate a POD status at step **446**. At step **448**, the updated order status is communicated to seller **44** (and may be directed to the specific warehouse which shipped the order, if appropriate).

The buyer at this stage can examine the contents of the shipment and communicate a message at step **450** identifying any discrepancy between the ASN and the actual delivery. DIMS **20** communicates any such message to seller **44** at step **452**. Furthermore, DIMS **20** may provides an exception management workflow to handle these exceptions. At step **454**, buyer **48** and seller **44** complete a financial settlement workflow in which the supplier communicates an invoice (for example, an EDI **810** message) to buyer **48** and buyer **48** makes a payment to seller **44** (for example, using an EDI **820** message), and the example method ends. In particular

embodiments, these financial settlement communication messages may be communicated between buyer **48** and seller **44** using DIMS **20**.

Although the present invention has been described with several embodiments, numerous changes, substitutions, variations, alterations, and modifications may be suggested to one skilled in the art, and it is intended that the invention encompass all such changes, substitutions, variations, alterations, and modifications as fall within the spirit and scope of the appended claims.

What is claimed is:

1. A distributed inventory management system, the system comprising one or more components collectively operable to:

receive information regarding a plurality of distributed participants in one or more tiers of a value chain, wherein each tier of the one or more tiers of the value chain comprise a different replenishment process associated with one or more of the plurality of distributed participants, and information regarding one or more items in the one or more tiers of the value chain;

model relationships between at least two of the plurality of distributed participants in the different tiers of the one or more tiers of the value chain based on the received information;

model the one or more items based on the received information;

receive inventory data from one or more of the plurality of distributed participants relating to the one or more items;

process the inventory data based on the models of the relationships and the items to generate inventory information related to one or more of the distributed participants; and

communicate the generated inventory information to one or more of the plurality of distributed participants in a different tier of the value chain, the inventory information including at least inventory data of participants other than those participants to which the inventory information is communicated.

2. The system of claim **1**, wherein the components are further operable to:

receive the inventory data from a plurality of different data management systems associated with the participants; and

translate the received inventory data for processing if such inventory data has a format which the components of the distributed inventory management system are unable to process.

3. The system of claim **2**, wherein one or more of the data management systems comprise an enterprise resource planning system, a warehouse management system, or a decision support system.

4. The system of claim **1**, wherein modeling the relationships between two or more of the participants comprises specifying the identity of one or more of the participants, the locations of the participants, the roles of the participants in the one or more tiers of the value chain, and one or more business rules associated with one or more of the participants.

5. The system of claim **1**, wherein modeling the items comprises specifying the identity of one or more items relevant in the one or more tiers of the value chain, a description of the items, the locations at which each item is active, and an item hierarchy.

6. The system of claim **1**, wherein the generated inventory information comprises the output of one or more business workflows that use an aggregation of the inventory data received from two or more of the participants.

7. The system of claim **1**, wherein the generated inventory information comprises a view of particular inventory data of a one or more first participants that is relevant to a second participant.

8. The system of claim **1**, wherein the generated inventory information comprises one or more alerts generated in response to the analysis of particular received inventory data according to one or more business rules.

9. The system of claim **1**, wherein the generated inventory information comprises one or more alerts generated in response to monitoring one or more key performance indicators.

10. The system of claim **1**, wherein the components are operable to communicate the generated inventory information to applicable participants upon generation of the inventory information.

11. The system of claim **1**, wherein the components are operable to communicate the generated inventory information to an applicable participant upon receiving a request from the participant.

12. The system of claim **1**, wherein the components are further operable to:

communicate inventory data associated with a plurality of the participants to one or more planning or scheduling engines;

receive inventory planning or scheduling information from the engines; and

communicate at least a portion of the planning information to one or more of the participants.

13. The system of claim **12**, wherein one or more of the engines comprise a replenishment planning engine, an order management system, a demand fulfillment engine, a supply chain planner, or a factory planner.

14. The system of claim **12**, wherein one or more of the engines are associated with one or more of the participants.

15. The system of claim **1**, wherein one or more of the components of the system are operated by one or more of the participants.

16. The system of claim **1**, wherein one or more of the components of the system are implemented in an e-marketplace independent of the participants.

17. A method for distributed inventory management, comprising:

receiving information regarding a plurality of distributed participants in one or more tiers of a value chain, wherein each tier of the one or more tiers of the value chain comprise a different replenishment process associated with one or more of the plurality of distributed participants, and information regarding one or more items in the one or more tiers of the value chain;

modeling relationships between at least two of the plurality of distributed participants in different tiers of the one or more tiers of the value chain based on the received information;

modeling the one or more items based on the received information;

receiving inventory data from one or more of the plurality of distributed participants relating to the one or more items;

processing the inventory data based on the models of the relationships and the items to generate inventory information related to one or more of the distributed participants; and

communicating the generated inventory information to one or more of the plurality of distributed participants in a different tier of the value chain, the inventory information including at least inventory data of participants

other than those participants to which the inventory information is communicated.

18. The method of claim **17**, further comprising:

receiving the inventory data from a plurality of different data management systems associated with the participants; and

translating the received inventory data for processing if such inventory data has a format which the components of the distributed inventory management system are unable to process.

19. The method of claim **18**, wherein one or more of the data management systems comprise an enterprise resource planning system, a warehouse management system, or a decision support system.

20. The method of claim **17**, wherein modeling the relationships between two or more of the participants comprises specifying the identity of one or more of the participants, the locations of the participants, the roles of the participants in the one or more tiers of the value chain, and one or more business rules associated with one or more of the participants.

21. The method of claim **17**, wherein modeling the items comprises specifying the identity of one or more items relevant in the one or more tiers of the value chain, a description of the items, the locations at which each item is active, and an item hierarchy.

22. The method of claim **17**, wherein the generated inventory information comprises the output of one or more business workflows that use an aggregation of 10 the inventory data received from two or more of the participants.

23. The method of claim **17**, wherein the generated inventory information comprises a view of particular inventory data of a one or more first participants that is relevant to a second participant.

24. The method of claim **17**, wherein the generated inventory information comprises one or more alerts generated in response to the analysis of particular received inventory data according to one or more business rules.

25. The method of claim **17**, wherein the generated inventory information comprises one or more alerts generated in response to monitoring one or more key performance indicators.

26. The method of claim **17**, the generated inventory information is communicated to applicable participants upon generation of the inventory information.

27. The method of claim **17**, the generated inventory information is communicated to an applicable participant upon receiving a request from the participant.

28. The method of claim **17**, further comprising:

communicating inventory data associated with a plurality of the participants to one or more planning or scheduling engines;

receiving inventory planning or scheduling information from the engines; and

communicating at least a portion of the inventory planning or scheduling information to one or more of the participants.

29. The method of claim **28**, wherein one or more of the engines comprise a replenishment planning engine, an order management system, a demand fulfillment engine, a supply chain planner, or a factory planner.

30. The method of claim **28**, wherein one or more of the engines are associated with one or more of the participants.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,574,383 B1
APPLICATION NO. : 10/120571
DATED : August 11, 2009
INVENTOR(S) : Parasnis et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

On the Title Page:

The first or sole Notice should read --

Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1024 days.

Signed and Sealed this

Fourteenth Day of December, 2010

David J. Kappos

Director of the United States Patent and Trademark Office

EXHIBIT 4

US007788145B2

(12) **United States Patent**
Wadawadigi et al.

(10) **Patent No.: US 7,788,145 B2**
(45) **Date of Patent: *Aug. 31, 2010**

(54) **INTELLIGENT FULFILLMENT AGENTS**

(75) Inventors: **Ganesh Wadawadigi**, Flower Mound, TX (US); **Ajay Jain**, Flower Mound, TX (US); **Deepak Mohapatra**, Richardson, AZ (US); **Sivakumar Balakrishnan**, Flower Mound, TX (US)

(73) Assignee: **i2 Technologies US, Inc.**, Dallas, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 156 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **11/877,087**

(22) Filed: **Oct. 23, 2007**

(65) **Prior Publication Data**

US 2008/0040245 A1 Feb. 14, 2008

Related U.S. Application Data

(63) Continuation of application No. 10/119,990, filed on Apr. 10, 2002, now Pat. No. 7,376,600.

(60) Provisional application No. 60/283,448, filed on Apr. 11, 2001.

(51) **Int. Cl.**

G06F 17/50	(2006.01)
G06F 9/44	(2006.01)
G06Q 10/00	(2006.01)

(52) **U.S. Cl.** **705/28**; 705/7; 705/10; 705/26; 700/233; 700/236

(58) **Field of Classification Search** 705/7, 705/8, 10, 14, 22, 28, 26, 27, 29; 700/233, 700/236

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,009,407 A * 12/1999 Garg 705/10

(Continued)

FOREIGN PATENT DOCUMENTS

JP 09288752 A 11/1997

OTHER PUBLICATIONS

Pfohl, Hans-Christian Cullmann, Oliver STolzle, Wolfgang, "Inventory Management with statistical process control: Simulation and evaluation", published by Journal of Business Logistics, v20nl, p. 100-200, 1999.*

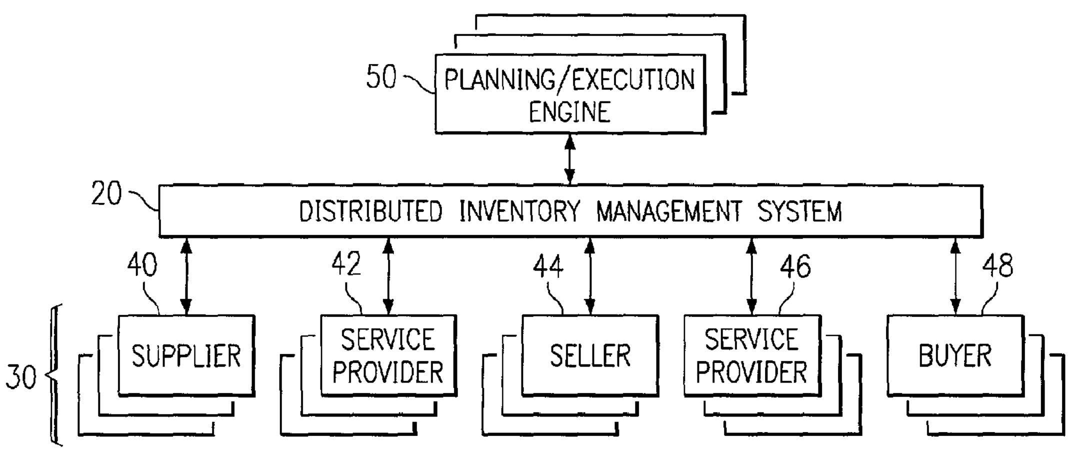
(Continued)

Primary Examiner—Matthew S Gart
Assistant Examiner—Olusegun Goyea
(74) *Attorney, Agent, or Firm*—Booth Udall, PLC; Steven J. Laureanti

(57) **ABSTRACT**

A method for distributed inventory management includes receiving information regarding a number of participants in a value chain and information regarding one or more items relevant in the value chain. The method also includes modeling relationships between two or more of the participants based on the received information and modeling the one or more items based on the received information. The method further includes receiving inventory data from the participants relating to the one or more items, evaluating the received inventory data according to one or more business rules associated with an agent, executing a business process associated with the agent based on the evaluation of the inventory data and the models of the relationships and the items, and communicating output of the business process to one or more of the participants.

20 Claims, 6 Drawing Sheets



U.S. PATENT DOCUMENTS

6,324,522	B2 *	11/2001	Peterson et al.	705/28
6,996,538	B2 *	2/2006	Lucas	705/28
7,376,600	B1 *	5/2008	Wadawadigi et al.	705/28
2001/0051905	A1 *	12/2001	Lucas	705/29
2002/0010633	A1 *	1/2002	Brotherston	705/26
2002/0138316	A1 *	9/2002	Katz et al.	705/7
2002/0143669	A1 *	10/2002	Scheer	705/28
2002/0188499	A1 *	12/2002	Jenkins et al.	705/10
2004/0236639	A1 *	11/2004	Candadai et al.	705/27
2005/0177435	A1 *	8/2005	Lidow	705/22
2005/0197929	A1 *	9/2005	Lucas	705/28

OTHER PUBLICATIONS

Andersson, J, Marklund, J, "Decentralized Inventory Control in a Two-Level Distribution System", published by European Journal of Operational Research, vol. 127, No. 3, p. 483-506, 2000.*

U.S. Appl. No. 10/120,571, entitled, "Distributed Inventory Management", 44 pages, filed Apr. 10, 2002.

Pfohl, Hans-Christian Cullmann, Oliver Stolzle, Wolfgang, "Inventory Management with statistical process control: Simulation and evaluation", published by Journal of Business Logistics, v. 20n1, pp. 100-200, 1999.

* cited by examiner

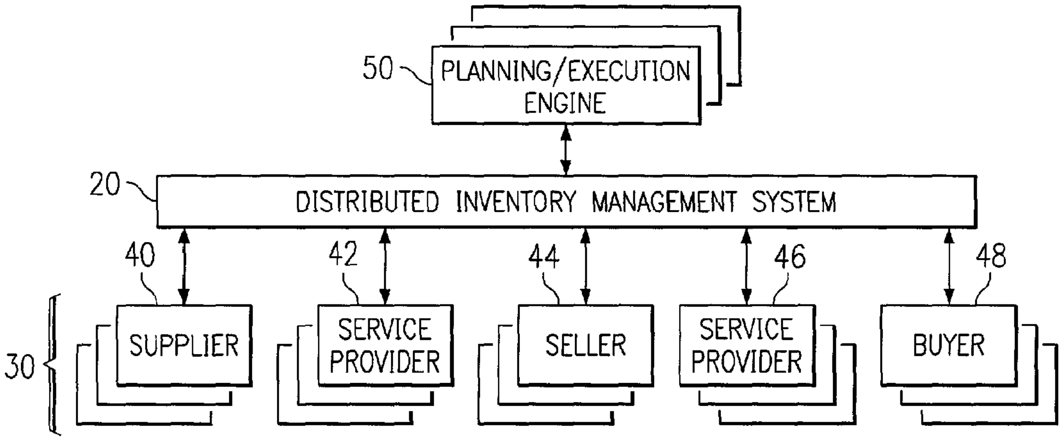

50
PLANNING/EXECUTION ENGINE
20
DISTRIBUTED INVENTORY MANAGEMENT SYSTEM
40
42
44
46
48
30
SUPPLIER
SERVICE PROVIDER
SELLER
SERVICE PROVIDER
BUYER

FIG. 1

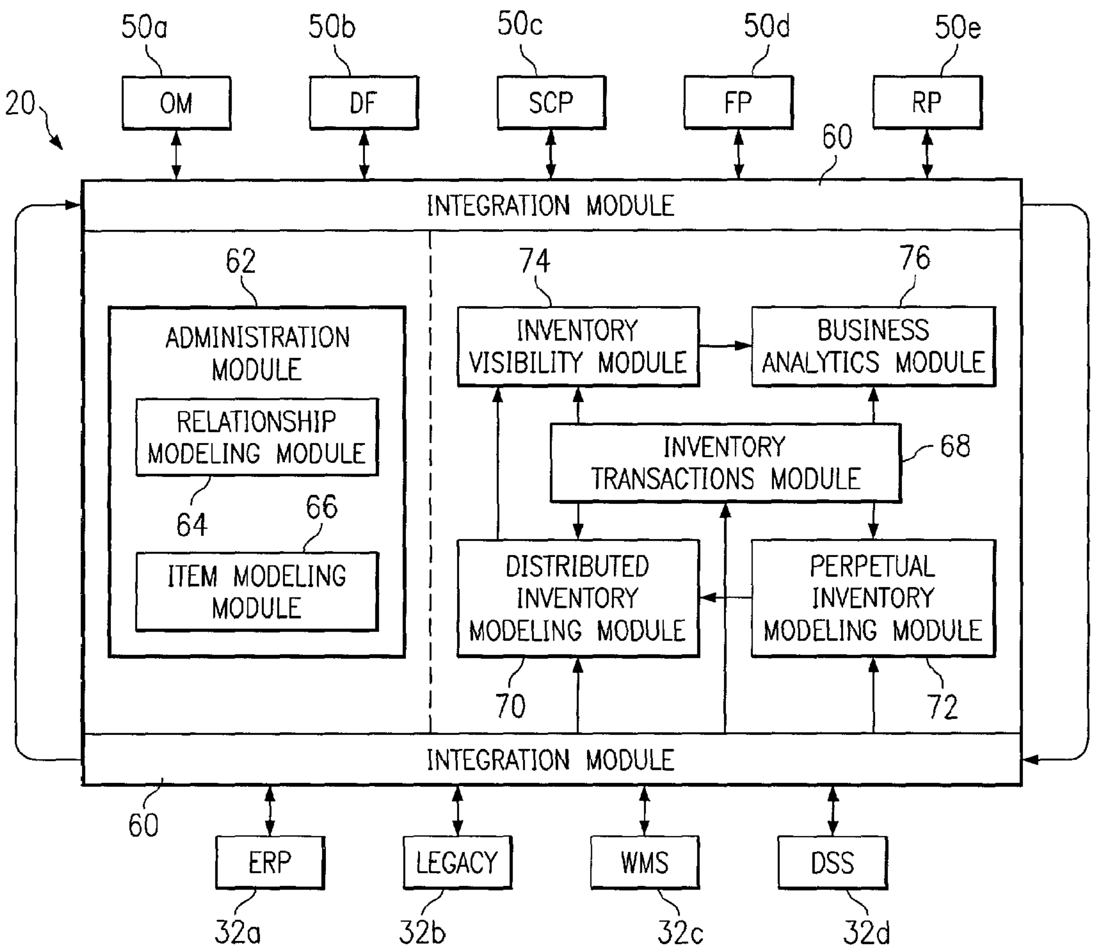

50a
50b
50c
50d
50e
20
OM
DF
SCP
FP
RP
60
INTEGRATION MODULE
62
ADMINISTRATION MODULE
RELATIONSHIP MODELING MODULE
64
66
ITEM MODELING MODULE
74
76
INVENTORY VISIBILITY MODULE
BUSINESS ANALYTICS MODULE
INVENTORY TRANSACTIONS MODULE
68
DISTRIBUTED INVENTORY MODELING MODULE
PERPETUAL INVENTORY MODELING MODULE
70
72
INTEGRATION MODULE
60
ERP
LEGACY
WMS
DSS
32a
32b
32c
32d

FIG. 2

FIG. 3

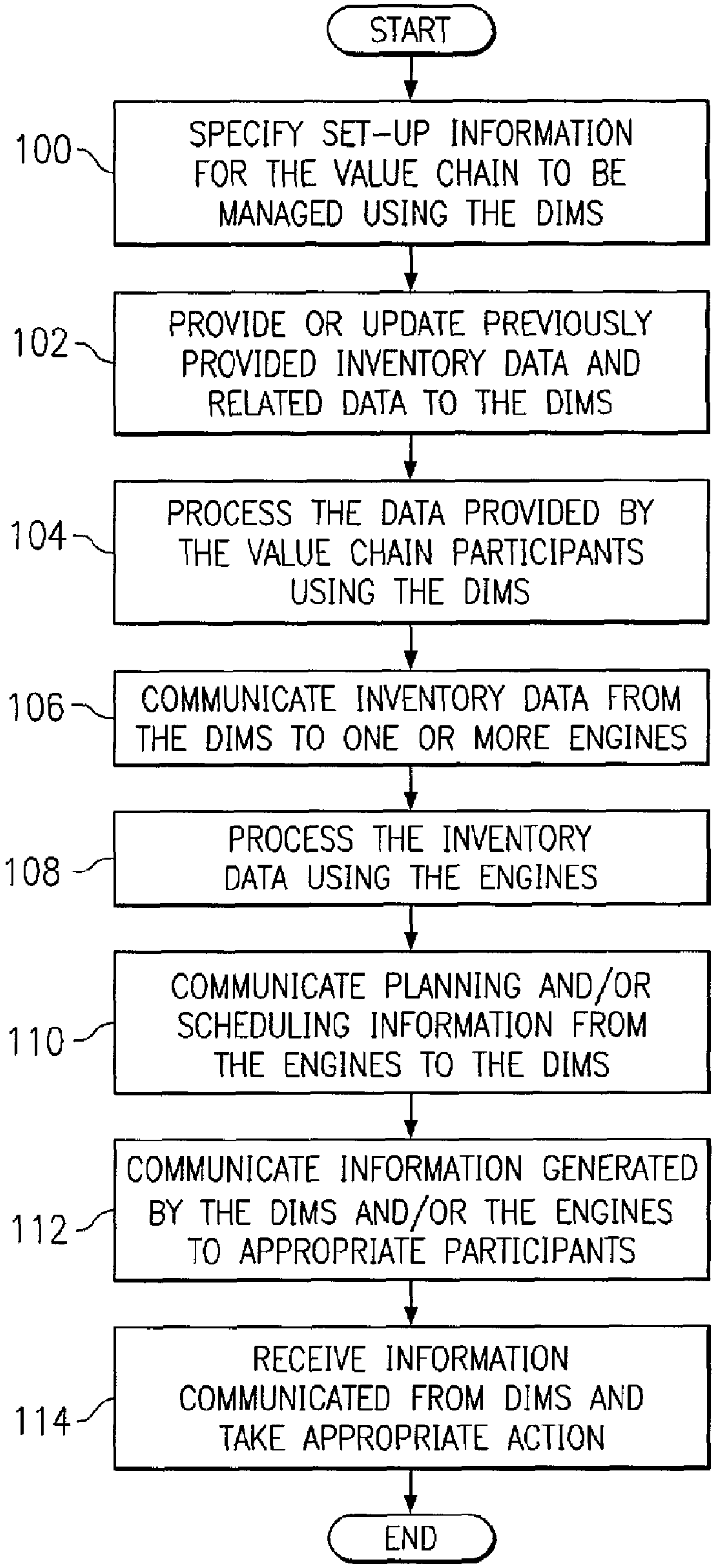

START
100
SPECIFY SET-UP INFORMATION FOR THE VALUE CHAIN TO BE MANAGED USING THE DIMS
102
PROVIDE OR UPDATE PREVIOUSLY PROVIDED INVENTORY DATA AND RELATED DATA TO THE DIMS
104
PROCESS THE DATA PROVIDED BY THE VALUE CHAIN PARTICIPANTS USING THE DIMS
106
COMMUNICATE INVENTORY DATA FROM THE DIMS TO ONE OR MORE ENGINES
108
PROCESS THE INVENTORY DATA USING THE ENGINES
110
COMMUNICATE PLANNING AND/OR SCHEDULING INFORMATION FROM THE ENGINES TO THE DIMS
112
COMMUNICATE INFORMATION GENERATED BY THE DIMS AND/OR THE ENGINES TO APPROPRIATE PARTICIPANTS
114
RECEIVE INFORMATION COMMUNICATED FROM DIMS AND TAKE APPROPRIATE ACTION
END

FIG. 4

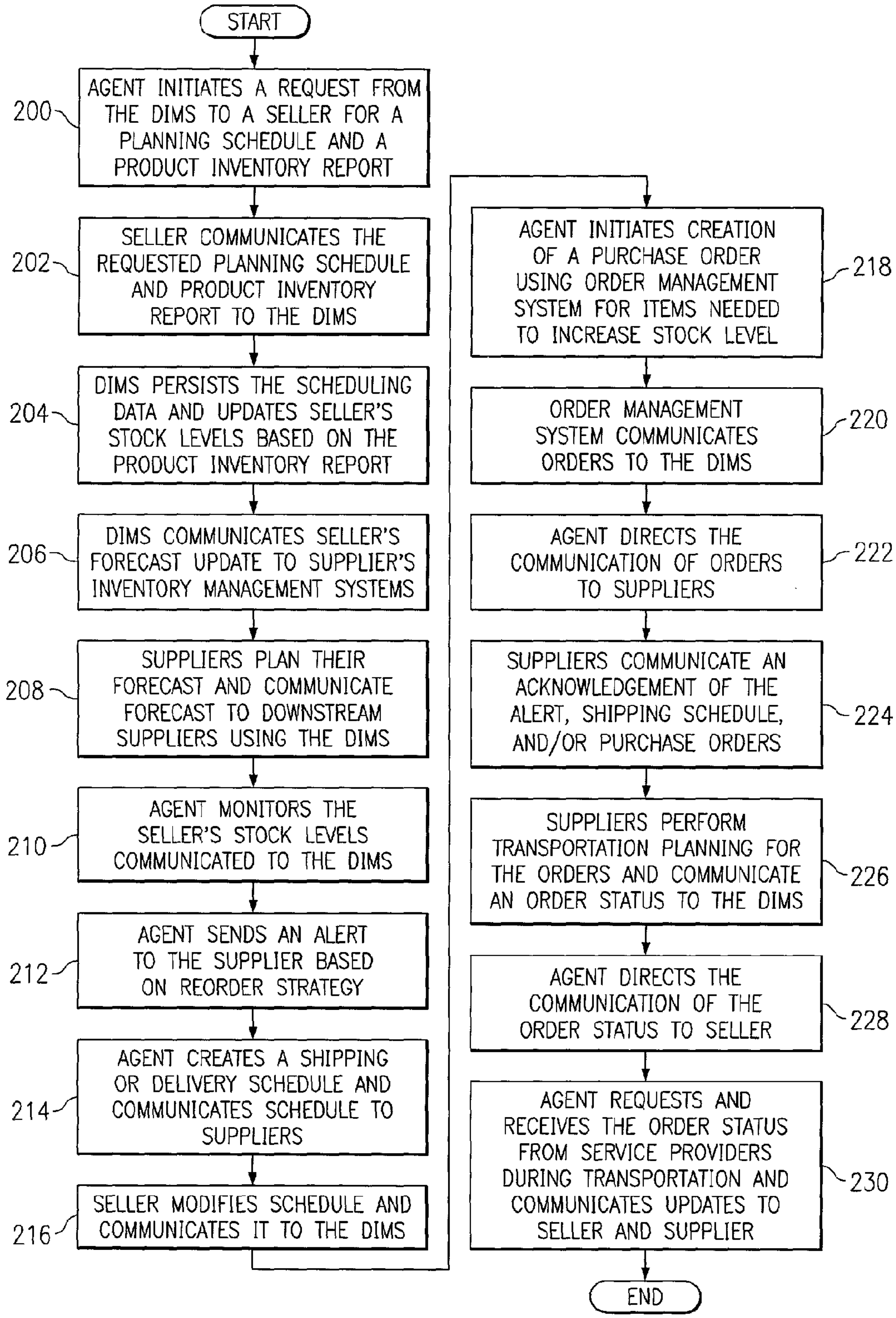

START
AGENT INITIATES A REQUEST FROM THE DIMS TO A SELLER FOR A PLANNING SCHEDULE AND A PRODUCT INVENTORY REPORT
200
SELLER COMMUNICATES THE REQUESTED PLANNING SCHEDULE AND PRODUCT INVENTORY REPORT TO THE DIMS
202
DIMS PERSISTS THE SCHEDULING DATA AND UPDATES SELLER'S STOCK LEVELS BASED ON THE PRODUCT INVENTORY REPORT
204
DIMS COMMUNICATES SELLER'S FORECAST UPDATE TO SUPPLIER'S INVENTORY MANAGEMENT SYSTEMS
206
SUPPLIERS PLAN THEIR FORECAST AND COMMUNICATE FORECAST TO DOWNSTREAM SUPPLIERS USING THE DIMS
208
AGENT MONITORS THE SELLER'S STOCK LEVELS COMMUNICATED TO THE DIMS
210
AGENT SENDS AN ALERT TO THE SUPPLIER BASED ON REORDER STRATEGY
212
AGENT CREATES A SHIPPING OR DELIVERY SCHEDULE AND COMMUNICATES SCHEDULE TO SUPPLIERS
214
SELLER MODIFIES SCHEDULE AND COMMUNICATES IT TO THE DIMS
216
AGENT INITIATES CREATION OF A PURCHASE ORDER USING ORDER MANAGEMENT SYSTEM FOR ITEMS NEEDED TO INCREASE STOCK LEVEL
218
ORDER MANAGEMENT SYSTEM COMMUNICATES ORDERS TO THE DIMS
220
AGENT DIRECTS THE COMMUNICATION OF ORDERS TO SUPPLIERS
222
SUPPLIERS COMMUNICATE AN ACKNOWLEDGEMENT OF THE ALERT, SHIPPING SCHEDULE, AND/OR PURCHASE ORDERS
224
SUPPLIERS PERFORM TRANSPORTATION PLANNING FOR THE ORDERS AND COMMUNICATE AN ORDER STATUS TO THE DIMS
226
AGENT DIRECTS THE COMMUNICATION OF THE ORDER STATUS TO SELLER
228
AGENT REQUESTS AND RECEIVES THE ORDER STATUS FROM SERVICE PROVIDERS DURING TRANSPORTATION AND COMMUNICATES UPDATES TO SELLER AND SUPPLIER
230
END

FIG. 5A

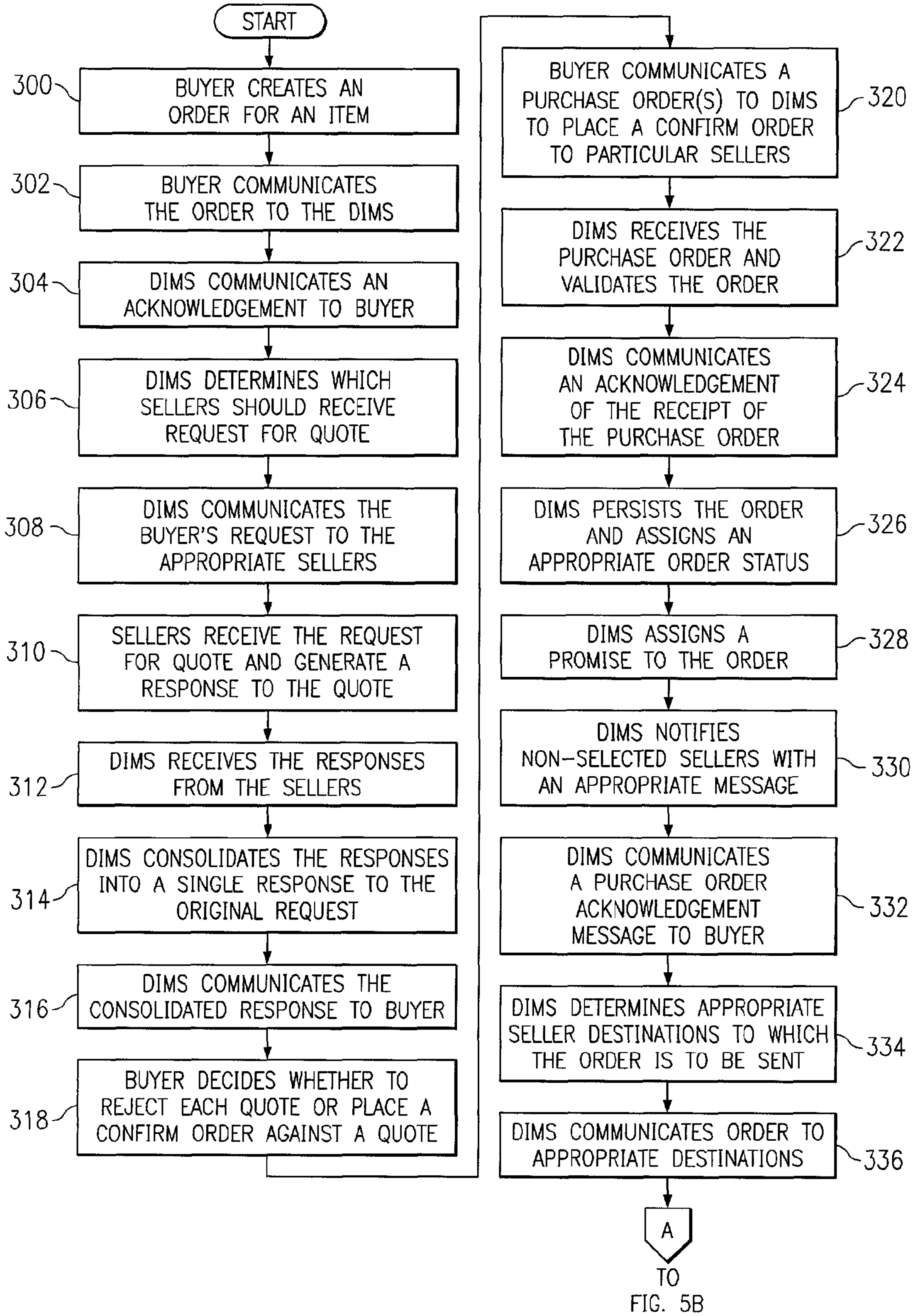

START
300
BUYER CREATES AN ORDER FOR AN ITEM
302
BUYER COMMUNICATES THE ORDER TO THE DIMS
304
DIMS COMMUNICATES AN ACKNOWLEDGEMENT TO BUYER
306
DIMS DETERMINES WHICH SELLERS SHOULD RECEIVE REQUEST FOR QUOTE
308
DIMS COMMUNICATES THE BUYER'S REQUEST TO THE APPROPRIATE SELLERS
310
SELLERS RECEIVE THE REQUEST FOR QUOTE AND GENERATE A RESPONSE TO THE QUOTE
312
DIMS RECEIVES THE RESPONSES FROM THE SELLERS
314
DIMS CONSOLIDATES THE RESPONSES INTO A SINGLE RESPONSE TO THE ORIGINAL REQUEST
316
DIMS COMMUNICATES THE CONSOLIDATED RESPONSE TO BUYER
318
BUYER DECIDES WHETHER TO REJECT EACH QUOTE OR PLACE A CONFIRM ORDER AGAINST A QUOTE
320
BUYER COMMUNICATES A PURCHASE ORDER(S) TO DIMS TO PLACE A CONFIRM ORDER TO PARTICULAR SELLERS
322
DIMS RECEIVES THE PURCHASE ORDER AND VALIDATES THE ORDER
324
DIMS COMMUNICATES AN ACKNOWLEDGEMENT OF THE RECEIPT OF THE PURCHASE ORDER
326
DIMS PERSISTS THE ORDER AND ASSIGNS AN APPROPRIATE ORDER STATUS
328
DIMS ASSIGNS A PROMISE TO THE ORDER
330
DIMS NOTIFIES NON-SELECTED SELLERS WITH AN APPROPRIATE MESSAGE
332
DIMS COMMUNICATES A PURCHASE ORDER ACKNOWLEDGEMENT MESSAGE TO BUYER
334
DIMS DETERMINES APPROPRIATE SELLER DESTINATIONS TO WHICH THE ORDER IS TO BE SENT
336
DIMS COMMUNICATES ORDER TO APPROPRIATE DESTINATIONS
A
TO FIG. 5B

FIG. 5B

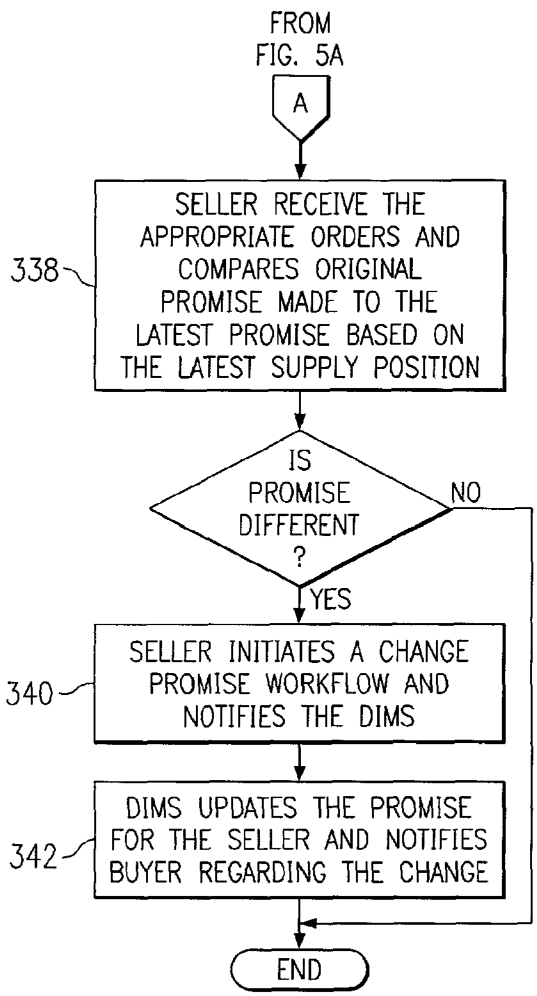

FROM FIG. 5A
A
338
SELLER RECEIVE THE APPROPRIATE ORDERS AND COMPARES ORIGINAL PROMISE MADE TO THE LATEST PROMISE BASED ON THE LATEST SUPPLY POSITION
IS PROMISE DIFFERENT ?
NO
YES
340
SELLER INITIATES A CHANGE PROMISE WORKFLOW AND NOTIFIES THE DIMS
342
DIMS UPDATES THE PROMISE FOR THE SELLER AND NOTIFIES BUYER REGARDING THE CHANGE
END

FIG. 6A

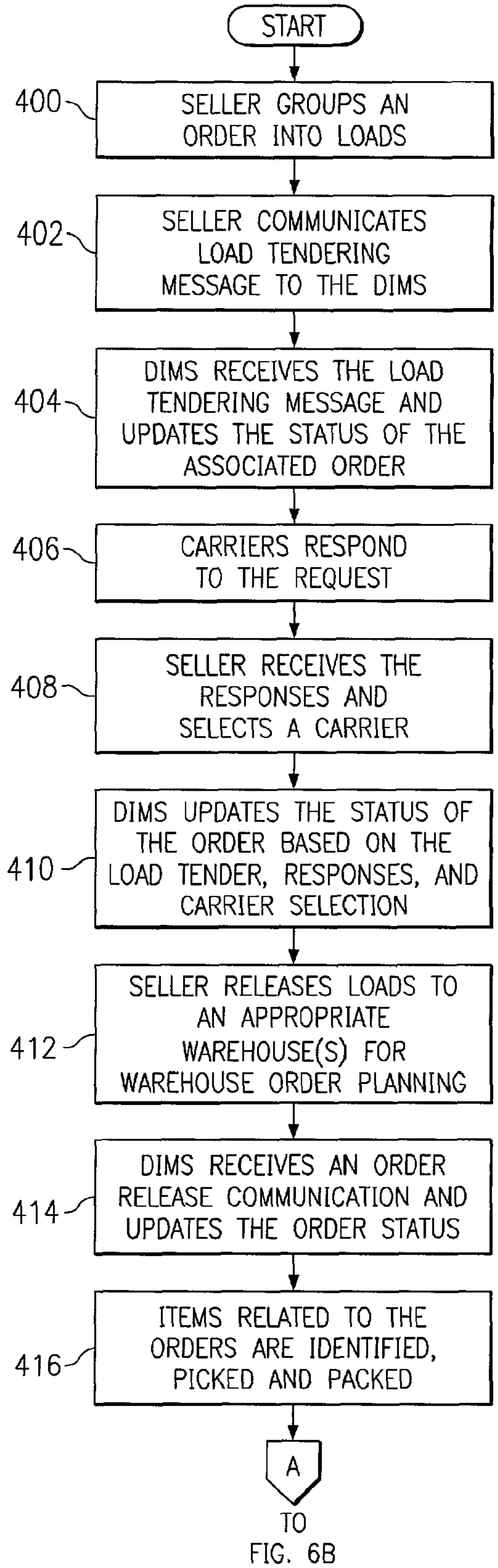
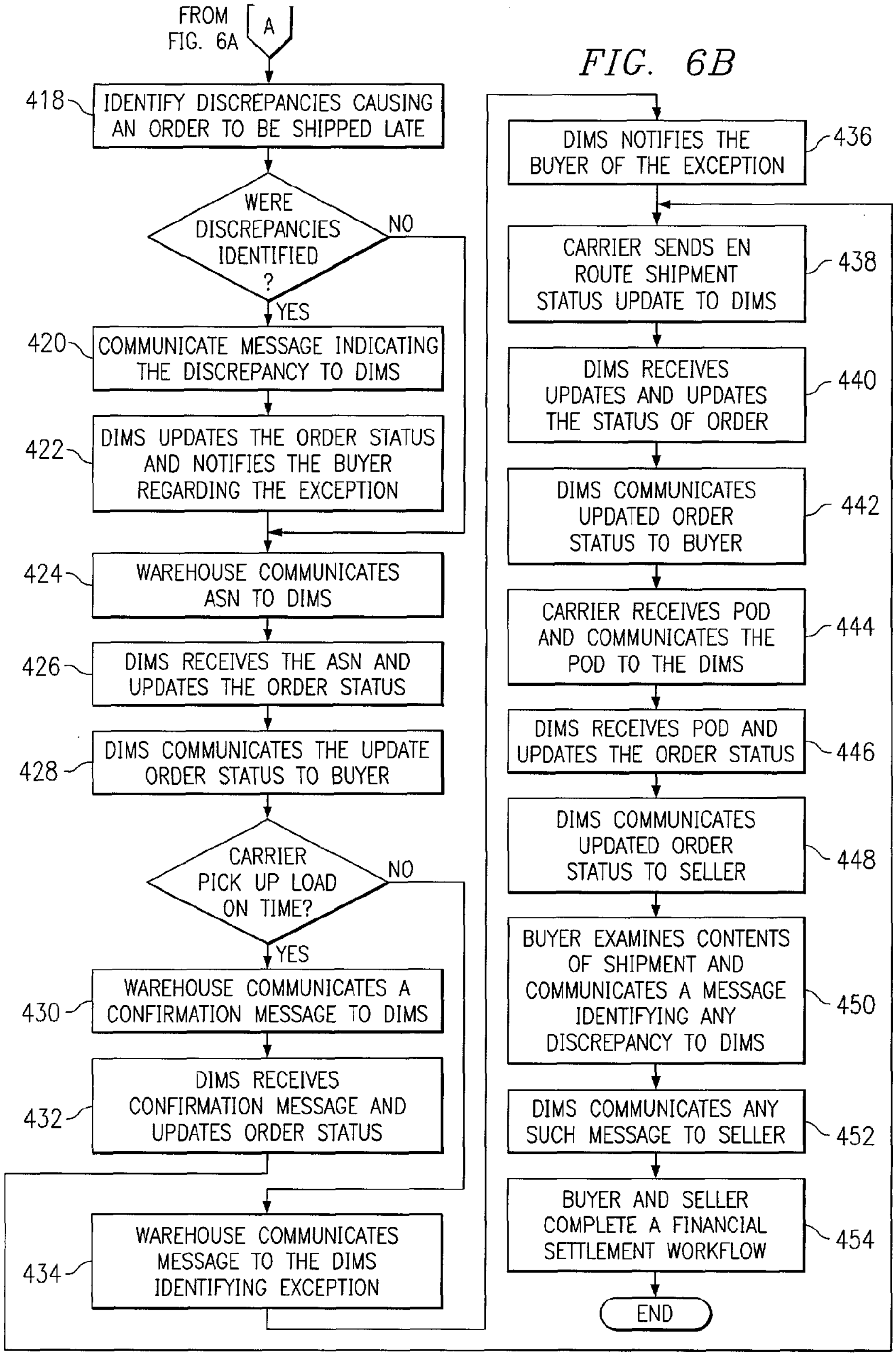

START
400
SELLER GROUPS AN ORDER INTO LOADS
402
SELLER COMMUNICATES LOAD TENDERING MESSAGE TO THE DIMS
404
DIMS RECEIVES THE LOAD TENDERING MESSAGE AND UPDATES THE STATUS OF THE ASSOCIATED ORDER
406
CARRIERS RESPOND TO THE REQUEST
408
SELLER RECEIVES THE RESPONSES AND SELECTS A CARRIER
410
DIMS UPDATES THE STATUS OF THE ORDER BASED ON THE LOAD TENDER, RESPONSES, AND CARRIER SELECTION
412
SELLER RELEASES LOADS TO AN APPROPRIATE WAREHOUSE(S) FOR WAREHOUSE ORDER PLANNING
414
DIMS RECEIVES AN ORDER RELEASE COMMUNICATION AND UPDATES THE ORDER STATUS
416
ITEMS RELATED TO THE ORDERS ARE IDENTIFIED, PICKED AND PACKED
A
TO FIG. 6B

FIG. 6B


FROM FIG. 6A
A
418
IDENTIFY DISCREPANCIES CAUSING AN ORDER TO BE SHIPPED LATE
WERE DISCREPANCIES IDENTIFIED ?
NO
YES
420
COMMUNICATE MESSAGE INDICATING THE DISCREPANCY TO DIMS
422
DIMS UPDATES THE ORDER STATUS AND NOTIFIES THE BUYER REGARDING THE EXCEPTION
424
WAREHOUSE COMMUNICATES ASN TO DIMS
426
DIMS RECEIVES THE ASN AND UPDATES THE ORDER STATUS
428
DIMS COMMUNICATES THE UPDATE ORDER STATUS TO BUYER
CARRIER PICK UP LOAD ON TIME?
NO
YES
430
WAREHOUSE COMMUNICATES A CONFIRMATION MESSAGE TO DIMS
432
DIMS RECEIVES CONFIRMATION MESSAGE AND UPDATES ORDER STATUS
434
WAREHOUSE COMMUNICATES MESSAGE TO THE DIMS IDENTIFYING EXCEPTION
DIMS NOTIFIES THE BUYER OF THE EXCEPTION
436
CARRIER SENDS EN ROUTE SHIPMENT STATUS UPDATE TO DIMS
438
DIMS RECEIVES UPDATES AND UPDATES THE STATUS OF ORDER
440
DIMS COMMUNICATES UPDATED ORDER STATUS TO BUYER
442
CARRIER RECEIVES POD AND COMMUNICATES THE POD TO THE DIMS
444
DIMS RECEIVES POD AND UPDATES THE ORDER STATUS
446
DIMS COMMUNICATES UPDATED ORDER STATUS TO SELLER
448
BUYER EXAMINES CONTENTS OF SHIPMENT AND COMMUNICATES A MESSAGE IDENTIFYING ANY DISCREPANCY TO DIMS
450
DIMS COMMUNICATES ANY SUCH MESSAGE TO SELLER
452
BUYER AND SELLER COMPLETE A FINANCIAL SETTLEMENT WORKFLOW
454
END

INTELLIGENT FULFILLMENT AGENTS

CLAIM OF PRIORITY

This application is a continuation of U.S. patent application Ser. No. 10/119,990, filed on 10 Apr. 2002 and entitled "INTELLIGENT FULFILLMENT AGENTS", which claims the benefit under 35 U.S.C. §119(e) of U.S. Provisional Application Ser. No. 60/283,448, filed 11 Apr. 2001 and entitled "SYSTEM INCORPORATING DISTRIBUTED INVENTORY BACKPLANE, INTELLIGENT COLLABORATIVE FULFILLMENT AGENTS, AND DISTRIBUTED ORDER FULFILLMENT CONCEPTS, SINGLY OR IN ANY COMBINATION". U.S. patent application Ser. No. 10/119,990 and U.S. Provisional Application Ser. No. 60/283,448 are commonly assigned to the assignee of the present application. The disclosure of related U.S. patent application Ser. No. 10/119,990 and U.S. Provisional Application Ser. No. 60/283,448 are hereby incorporated by reference into the present disclosure as if fully set forth herein.

BACKGROUND

1. Technical Field of the Invention

This invention relates generally to the field of business management and more particularly to intelligent fulfillment agents.

2. Background of the Invention

Various inventory replenishment processes are employed by businesses to fulfill orders and manage inventory. Although business use various types of replenishment processes (which are often customized for the business), some of the most commonly used processes are Vendor-Managed Inventory (VMI) programs, Supplier-Managed Inventory (SMI) programs and Just in Time (JIT) programs. In a VMI program, a vendor of products takes over the responsibility of managing the inventory of certain products for a given customer. Depending on the situation, the vendor might receive data such as forecasted demand, product consumption rates, inventory positions from the customer, and other inventory-related data. The vendor may also or alternatively be responsible for generating part of the data on the customers behalf. The vendor then uses the received and/or generated data to determine how much of the product to replenish and when such replenishment should occur.

In an SMI program, a seller negotiates with a sub-set of its suppliers to create supplier hubs dedicated to that seller (and which may be managed by a third party). As part of the negotiation, the third party and the suppliers agree to particular service levels at a hub, such as speed of delivery of replenishments of inventory and minimum and maximum stock levels. On a periodic basis throughout a given day, the seller may communicate with the third party provider to move material from the hub to the plant (a JIT arrangement). On a weekly or monthly basis, the seller may send a forecast to the suppliers. The suppliers are then responsible for maintaining inventory within the negotiated levels at the hub using this daily product movement and forecast information.

Although these various replenishment processes provide adequate inventory management, problems arise when multiple entities are involved in a value chain and especially when these multiple entities carry out different types of replenishment programs between one another.

SUMMARY OF THE INVENTION

According to the present invention, disadvantages and problems associated with previous business management systems have been substantially reduced or eliminated.

According to one embodiment of the present invention, a method for distributed inventory management includes receiving information regarding a number of participants in a value chain and information regarding one or more items relevant in the value chain. The method also includes modeling relationships between two or more of the participants based on the received information and modeling the one or more items based on the received information. The method further includes receiving inventory data from the participants relating to the one or more items, evaluating the received inventory data according to one or more business rules associated with an agent, executing a business process associated with the agent based on the evaluation of the inventory data and the models of the relationships and the items, and communicating output of the business process to one or more of the participants.

Particular embodiments of the present invention may provide one or more technical advantages. For example, certain embodiments of the present invention real-time visibility into the inventory information associated with distributed participants in multiple tiers of a value chain. Particular embodiments also integrate these participants and their inventory information with various decision support planning and execution systems and provide business analysis with features such as monitoring of key performance measures (such as order fill rates, shipment variance, and inventory turns), reporting, and exception and alert management. Certain embodiments also provide integration into planning, order management and warehouse management systems of value chain participants.

Embodiments of the present invention may also provide improvements over existing systems, such as reduced capital blocked in inventory (resulting in improved return on assets), reduced warehouse and handling costs, reduced cost of goods sold, increased inventory turns, reduced logistics costs, decreased exposure to price protection liabilities, and increased customer satisfaction.

Other important technical advantages are readily apparent to those skilled in the art from the figures, descriptions and claims included herein.

BRIEF DESCRIPTION OF THE DRAWINGS

To provide a more complete understanding of the present invention and the features and advantages thereof, reference is made to the following description taken in conjunction with the accompanying drawings, in which:

FIG. **1** illustrates an example distributed inventory management system for managing the inventory of one or more participants in a value chain;

FIG. **2** illustrates an example distributed inventory management system in further detail;

FIG. **3** illustrates an example method for replenishment planning and execution using a distributed inventory management system;

FIG. **4** illustrates an example inventory workflow that may be implemented and automated using agents;

FIGS. **5**A and **5**B illustrate an example distributed order fulfillment method; and

FIGS. **6**A and **6**B illustrates an example distributed order execution method.

DETAILED DESCRIPTION OF THE DRAWINGS

FIG. **1** illustrates an example distributed inventory management system (DIMS) **20** for managing the inventory of one or more participants **30** in a value chain. DIMS **20** couples inventory data resources of a number of different participants **30** in a value chain with one or more planning and/or execution engines **50** that use the inventory data to plan and execute inventory-related business decisions. Participants **30** may include one or more suppliers **40** of components used to manufacture or otherwise create products, one or more sellers **44** that manufacture or otherwise create the products (for example, a seller **44** may be an original equipment manufacturer), and one or more buyers **48** that purchase the products from sellers **44**. Participants **30** may also include one or more service providers **42** and **46** that serve as intermediaries between suppliers **40** or buyers **48** and sellers **44**. As examples only, service providers **42** may include third party logistics (3PL) providers, contract sellers, or distributors, and service providers **46** may include 3PL providers, value-added resellers (VARs), or distributors.

It should be noted that each participant **30** may act as a "buyer" or a "seller" with respect to another entity in the value chain and the terms used above to name each participant **30** should not be construed as limiting the roles of participants **30**. Furthermore, participants may include one or more sub-organizations that carry out the various roles. As an example only, a seller **44** may be a manufacturer that includes a buying sub-organization that interacts with suppliers **40** (either directly or through a service provider **42**) to obtain components, a manufacturing sub-organization that manufactures one or more products using the components, and a selling sub-organization that interacts with buyers **48** (either directly or through a service provider **46**) to sell the manufactured products. Participants **30** may alternatively or additionally include any other entities participating in a value chain. The present invention also contemplates that some or all of participants **30** may be associated with the same entity (for example, multiple participants **30** may be different divisions of a company).

In general, DIMS **20** brokers pertinent data between participants **30** and engines **50** to propagate inventory planning and execution information between participants. For example, participants **30** may communicate inventory data to DIMS **20** and DIMS **20** may process this data and communicate inventory information from the processed data back to one or more participants **30**. DIMS **20** may also or alternatively communicate this data from participants **30** to one or more engines **50**. Engines **50** may then communicate inventory planning and execution information back to DIMS **20**, which may then make this information available to some or all of participants **30**. Although DIMS **20** and engines **50** are illustrated in FIG. **1** as being separate from participants **30**, it should be understood that some or all of DIMS **20** may be associated with one or more of participants **30**. For example, some or all modules of DIMS **20**, as will be described below, may be associated with a seller **44** and the suppliers **40**, service providers **42** and **46**, and/or the buyers **48** who wish to share inventory planning and execution information amongst one another may communicate inventory data to and receive inventory information from the components of DIMS **20** associated with the seller **44**. As an alternative, DIMS **20** may be implemented independently from participants **30** (for example, in an ecommerce marketplace or other node of a trading network). In a similar manner, each engine **50** may be associated with a particular participant **30** or may be implemented independently from participants **30**. As another alternative, a DIMS **20** may exist at each of a number of locations (either associated with a participant **30** or a third party) and/or the components of an individual DIMS **20** may be distributed amongst multiple locations.

Participants **30** may each operate one or more computer systems at one or more locations. These systems of participants **30** may interact with DIMS **20** autonomously or according to input from one or more users associated with a participant **30**. DIMS **20**, the systems of participants **30**, and engines **50** may be coupled to one another using one or more local area networks (LANs), metropolitan area networks (MANs), wide area networks (WANs), a portion of the global computer network known as the Internet, or any other appropriate wireline, wireless, or other links. Participants **30** and DIMS **20** may be arranged and communicate with one another according to a hub-and-spoke, peer-to-peer, or any other suitable architecture.

The use of DIMS **20** for inventory management provides numerous advantages over previous techniques, which have inherent inefficiencies that are caused by a lack of connectivity, timely communication, and visibility between the various participants **30**. Although current techniques for implementing replenishment processes (such as VMI, SMI, and JIT) are suitable for the needs of some businesses, these techniques include several disadvantages. In many value chains, different replenishment programs and policies are used in different tiers of the value chain. For example, a JIT program may be used between a seller **44** and a service provider **42** and a separate SMI policy may be used between the service provider **42** and a supplier **40**. Because of the disconnect between the JIT policy used between the seller **44** and the service provider **42** and the SMI policy used between the service provider **42** and the supplier **40**, inefficiencies are created in the value chain. Furthermore, it is also common in value chains for a participant **30**, such as an seller **44**, to use different policies with different trading partners. For example, an seller **44** may use a VMI policy with one buyer **48** (for example, a retailer) and may use a non-VMI policy with another buyer **48**. As with the use of different policies in different tiers of the value chain, this disconnect in the policies in use between the seller **44** and the two buyers **48** also introduces inefficiencies.

In addition to problems associated with the use of different replenishment policies, problems also arise when attempting to obtain data from the various participants **30** as input to one or more planning or execution engines **50**, such as a replenishment planning engine. Such engines **50** can add significant value in the planning of activities and flows within the value chain; however, it has been problematic to gather all the data from the various participants **30** in a consistent fashion and to integrate the engines **50** with each participant **30**. Furthermore, in situations where planning or execution engines are run in a batch mode (often due to the lack of real-time data from participants **30**), participants **30** have not able to leverage the real-time feed-back of inventory changes occurring in the value chain. In short, most inventory management systems lack connectivity between participants **30** and do not provide adequate visibility into inventory information of the participants in the value chain. This leads to increased production costs and reduced profits through the value chain due to problems such as unanticipated stock-outs, decreased production efficiency, higher investment in safety stock, increased obsolete and excess inventory, and inventory being in the wrong locations in the value chain.

The use of DIMS **20** reduces or eliminates these inefficiencies and allows each participant **30** in a value chain to simultaneously lower inventory, improve responsiveness, and

lower costs. Additionally, participants **30** can collectively understand how relationships and interactions need to be modified in order to achieve or improve on desired objectives. DIMS **20** provides participants **30** with one logical distributed inventory "backplane" which has the capability to model engagement rules to enable interoperability of fulfillment/replenishment policies in different tiers of a value chain or co-existence of multiple fulfillment/replenishment policies for a given participant **30** in a value chain. This is accomplished by providing a framework to construct and maintain a virtual unification of the different systems against which business intelligence can be written.

Each participant **30** in a value chain typically generates exposed or exposable information relating to that participant's **30** physical elements within the value chain (such as warehouses, manufacturing facilities, stores, and delivery trucks). For example, this information may include information regarding which products are in which warehouse of a participant **30** or which orders map to which lots on a particular truck of a participant **30**. Each participant **30** also may create exposed or exposable business logic and operations at the element level of the supply chain (for example, purchase order acceptance and generation functions and order processing functions). However, this information is often not available throughout the value chain. DIMS **20** collects this information and business logic to provide a unified and normalized data model of the inventory information available at all participants **30** in the value chain that are coupled to and communicate with DIMS **20**. For example, the data model may identify which products are in which warehouses across the entire value chain or identify which orders map to which lots on which truck at any point in the entire value chain. Therefore, DIMS **20** may provide any participant **30** visibility, typically on a permissions basis, into inventory information of any other participant **30**. In addition, this unified and normalized information may be communicated to one or more engines **50**, and the output from the one or more engines **50** may be provided to any relevant participants **30** for planning and execution purposes.

FIG. **2** illustrates an example DIMS **20** in further detail. DIMS **20** includes several components or modules that perform various inventory management or related functions. DIMS **20** may be implemented as any appropriate combination of software and/or hardware operating in one or more locations. In one example embodiment, all of the modules associated with DIMS **20** are executed on one or more computers associated with a particular participant **30**. In another example embodiment, certain modules may be executed on one or more computers associated with a first participant **30**, other modules may be executed on one or more computers associated with a second participant **30**, and yet other modules may be executed on one or more computers associated with a third party (such as in an e-commerce marketplace). Furthermore, all of the modules may be executed on one or more computers associated with a third party. Any other appropriate location and distribution of the modules may also be used.

One module that the example DIMS **20** includes is an integration module **60**. Integration module **60** serves to integrate DIMS **20** with various information systems **32** of participants **30** and with one or more planning and execution engines **50**. This integration allows inventory data and other related information to be received and processed by DIMS **20** and, where appropriate, engines **50**. Integration module **60** allows information to be communicated between DIMS **20** and/or engines **50** to information system **32** or other systems associated with participants. Integration module **60** may handle connectivity to upload and download data in the context of the role-based or system-based workflows. The integration may be at a custom API level, as well as conforming to industry standards like EDI, RosettaNet™, and the like. Based on the characteristics of the various participants **30** that need to interact, integration module **60** may be configurable to interact on a transaction-by-transaction basis, a net change basis, using a complete batch refresh of data at specified time intervals, and/or on any other basis.

Integration module **60** may include appropriate components for communicating with and accessing data stored at or communicated from participant systems **32**. Systems **32** may include resource planning (ERP) systems **32***a*, legacy systems **32***b*, warehouse management systems (WMS) **32***c*, decision support systems (DSS) **32***d*, and/or any other appropriate type of systems operated by one or more participants **30** that are associated with the management of inventory and related information. Appropriate components of integration module **60** may receive or request inventory data from one or more systems **32** associated with one or more participants **30** and may then communicate this information to other modules of DIMS **20** and/or to engines **50** for processing. Therefore, integration module **60** may also include one or more components for communicating with and accessing data stored at or communicated from engines **50**. Engines **50** may include order management engines **50***a*, demand fulfillment engines **50***b*, supply chain planning engines **50***c*, factory planning engines **50***d*, replenishment planning engines **50***e*, and/or any other appropriate type of planning and/or execution engines that may be useful for the management of inventory and related information. As described above, these engines **50** may be associated with one or more participants **30** or with a third party (such as with an e-commerce marketplace).

Integration module **60** may also perform translation of data received from a system **32** and/or an engine **50** as appropriate for the destination to which the data is to be sent. For example, integration module **60** may translate the data format of inventory data received from an ERP system **32***a* so that it may be used by other modules of DIMS **20** and/or a replenishment planning system **50***e*. Planning data received from the replenishment planning system **50***e* may then be translated back to a format appropriate for the ERP system **32***a* and/or any other systems **32**. Integration module **60** may also or alternatively perform any other appropriate types of translation. Furthermore, integration module **60** may manage requests for data from systems **32** and/or engines **50** and responses to these requests. For example, integration module **60** may provide authentication and authorization of such requests. Integration module **60** may also serve to queue requests and responses, if appropriate, for communication to a suitable destination.

The example DIMS **20** also includes an administration module **62** that is used to configure and set-up the users of DIMS **20** which includes configuring the identity of the users of the system (participants **30** in a value chain) and their role-based permissibility as well as the frequency and types of messages that DIMS **20** is going to receive from participants **30** and other external sources. Two primary aspects of administration module **62** are a relationship modeling module **64** and an item modeling module **66**.

Relationship modeling module **64** is used to specify parameters associated with the inter- and intra-enterprise participants **30** in the value chain. The transactions and the data permissibility are governed by the rules configured using module **66**. Participants **30** that may be modeled include the various divisions of an enterprise and their locations (including location hierarchies) within the enterprise, as well as the customers and the suppliers that the enterprise interacts with.

In essence this is the physical map that lays out the various participants **30** with which DIMS **20** is to interact, as well as their locations, roles, business rules, and/or any other appropriate parameters. Module **66** may also be used to implement agents, as described below.

Item modeling module **66** is used to specify the items relevant in the value chain, their descriptions, the locations at which the items are active, the item hierarchy, the unit of measure (DOM) of each item (and any hierarchies for the DOMs), and/or any other suitable parameters relating to the items. Module **66** also plays a role in the aggregation of data, described below, in the sense that cross-referencing rules for the items may be specified using this module **66**. The ownership aspect of inventory is also specified using module **66**.

The example DIMS **20** also includes an inventory transactions module **68** that provides industry specific-configurable transaction mappings. The transactions may also have the capability of being solution-specific within an industry. All relevant transactions may be defined using module **68**. These transactions may then facilitate interaction with systems **32** and engines **50**. Module **68** also provides views of transactions as well as techniques for manually creating and editing transactions. A transaction history may also be maintained for audit trail purposes.

Furthermore, the example DIMS **20** includes an distributed inventory modeling module **70** and perpetual inventory modeling module **72**. Distributed inventory modeling module **70** provides a unified view of inventory information across the distributed systems **32** of participants **30** (which, as described above, may be intra-enterprise situation and/or inter-enterprise entities). Distributed inventory modeling module **70** provides a configurable and expandable inventory state map definition framework. This is useful since modeling a distributed environment typically requires interfaces with disparate types of systems **32**. Also, module **70** provides the ability to model computed and instance-specific states of inventory to support various industry solution workflows. Perpetual inventory modeling module **72** functions to act as a complete system supporting all the functions of a enterprise inventory management system. Perpetual inventory modeling module **72** functions to act as a complete system supporting all the functions of a enterprise inventory management system, such as transaction reconciliation, attribute level lot tracking, and tracing functions.

The example DIMS **20** also includes an inventory visibility module **74** that is integrated with inventory modeling modules **70** and **72** and inventory transaction module **68** to provide users with views into the distributed inventory environment. Module **74** also provides user-configurable reports that allow users with different roles to quickly access relevant data and make informed decisions. Based on information from modeling module **70**, a users can select a relevant portion of the inventory data regarding to view. Furthermore, the views may also provide information regarding the underlying transactions affecting the inventory in the value chain. Based on the permissibility framework established using relationship modeling module **64**, participants **30** may able to only view inventory which they have permission to view. Module **74** also provides the ability to define user-defined event triggers based on the distributed inventory information, thus enabling proactive management of inventory.

In addition, the example DIMS **20** includes a business analytics module **76** that provides the analytics services for both inventory transactions and for past, current and future inventory positions. For example, transaction audit trails may be used to view lot track and trace information. Furthermore, trending and profiling information may be provided using module **76**. For example, module **76** may be used to track the performance of new products as well as products that are being phased out, may be used to analyze liability and exposure in cases of consigned inventory, may be use to keep and track participant performance metrics, and/or may be used to perform any number of other suitable business analyses.

Although particular example modules have been described, it should be understood that DIMS **20** may include additional module, may not include some of the example modules, and may implement particular functions in a different manner than described above. The various modules of DIMS **20** collectively provide a common, distributed backplane between multiple participants **30** in different tiers of a value chain. This common backplane enables each participant **30** to acquire relevant information from other participants **30** related to inventory management and enables multiple participants **30** to provide data to a planning or execution engine **50** and receive the output of that engine **50** that is generated using the input of the multiple participants **30**. For example, an seller **44** is able to obtain information on planned shipments of components from suppliers **42** and information on planned purchases by buyers **48**, and thus is able to manage its inventory accordingly.

By enabling the common distributed backplane between the various participants **30**, a seller **44** (or any other participant **30**) is able to manage products being fulfilled and replenished using different policies through a single gateway. This provides the seller **44**, for example, with a consolidated and accurate status on the state of affairs of a particular product with respect to multiple buyers **48** even though different inventory management policies may be used with each buyer **48**. Service providers **42**, **46** can also be updated with respect to the planned movement of goods in the value chain. Therefore, all participants **30** potentially can have visibility into any appropriate information needed and can get this information from a single source, regardless of the inventory management program(s) in which the participant **30** is involved. This process creates efficiencies in the value chain by reducing or eliminating the need to plan against uncertainties created due to the lack of information related to inventory.

DIMS **20** also provides for different user roles (such as brand manager, fulfillment manager, and replenishment planner) that provide users with a customized view of inventory across various geographical and product hierarchies. For example, a brand manager might be only interested in getting visibility into safety stock levels across all locations and to be alerted to the possibility of a channel being starved or carrying excess. A replenishment planner, in addition to the information required by a brand manager, might also need visibility into in-transit inventory and an average weekly forecast. The use of DIMS **20** to model the value chain enables this custom visibility to different users. For example, geographic hierarchies can be modeled and thus a user of DIMS **20** can log on and request inventory information on a given product family for a given geographic region. In this case, inventory positions for that product family can be aggregated across all the warehouses in that region (and also from the stored connected to the warehouses, if this information is provided). The user would then be able to drill down to specific locations, and even get details by buyer and buyer location. DIMS **20** may also enable modeling of product hierarchies. In this case, for example, a brand manager can acquire visibility to inventory at a brand level and a replenishment planner can get visibility at a product stock keeping unit (SKU) level.

DIMS **20** also allows for easy definition of business rules throughout the value chain. Different users can easily monitor the value chain by receiving notifications of violations of the

business rules. These rules can be based on minimum or maximum quantity inventory positions, computed minimum or maximum days of inventory based on projected sales-out and ending inventory, or based on numerous other criteria. Upon violation of a rule, DIMS may communicate a notification to a designated list of users. Furthermore, a violation of a rule may trigger a business process. As an example only, in scenarios like VMI, a violation of a minimum quantity rule may result in an alert being communicated to relevant participants **30** and/or a replenishment order being generated using an order management system. The generation of such a notification and the initiation of business processes may be performed using agents, as described in further detail below.

Furthermore, business logic may exist in DIMS **20** that serves as an intelligent mediator between systems **32** and engines **50**. For example, the intervening business logic may: evaluate incoming changes to determine if re-planning or aborting of an ongoing planning cycle is warranted, reconcile plan results with changes made in the model after a planning snapshot is taken, compare plan results with the current model state to compute net changes for communication to the chain elements, and/or direct messaging or model modifications to apply the model in accordance with the computed plans. The resulting combination of DIMS **20**, batch mode replenishment planning engine(s) **50***e*, and the logic described above provides a quasi-adaptive replenishment planning solution. Such a solution provides benefits such as flagging of exceptions in mismatches when different policies are used in different tiers in the value chain, use of a common inventory backplane to manage products under different policies with different participants **30**, an efficient way to evolve or change from one policy to another, and an efficient way to evolve from a batch-based fulfillment/replenishment planning paradigm to an adaptive fulfillment/replenishment planning paradigm.

FIG. **3** illustrates an example method for replenishment planning and execution using DIMS **20**. Using DIMS **20**, automated replenishment planning and execution in the multi-party inventory chain (including some or all participants **30** of a value chain) can be addressed by using data collected from the multiple participants **30** in a replenishment planning engine **50***e* and/or by using business logic associated with DIMS **20** to communicate the resulting inventory plans to the relevant participants **30** in the inventory chain.

The example method begins at step **100** where one or more participants in the value chain communicate with administration module **62** to specify set-up information for the value chain to be managed. The set-up information provides DIMS **20** with information regarding the make-up and characteristics of the value chain so that DIMS **20** may be used to manage the value chain. As an example, set-up information provided by a supplier **40** may include, but is not limited to: information regarding products, product components, or other items included in the value chain to be managed and information regarding the participants **30** in the value chain (such as the identity and location of the various participants **30**, the role of participants **30**, the manner in which particular participants **30** interact, and business rules associated with participants **30**). Participants **30** may communicate with administration module **62** using any appropriate communication techniques.

At step **102**, participants **30** provide or update previously provided inventory data and related data (such as demand, supply, and capacity data) using systems **32** and/or other appropriate data sources. As an example, a seller **44** may update inventory planning data to reflect an excess or shortage at supplier **40**. Data is provided to DIMS **20** through integration module **60** which is operable to communicate with systems **32** and perform any necessary data translation for use by DIMS **20**. Data may be communicated to DIMS **20** on a as generated, as requested, on scheduled basis, or in any other appropriate manner. Furthermore, this communication of data may be based on a "push" or "pull" method. For example, systems **32** may communicate or "push" data to DIMS **20** when relevant data is available or on a scheduled basis. Alternatively or in addition, DIMS **20** may request or "pull" relevant data from systems **32** as needed or on a scheduled basis. DIMS **20** may store the data in any appropriate data storage location associated with or independent of one or more of the modules of DIMS **20**.

At step **104**, one or more modules of DIMS **20** perform internal processing of the data provided by participants **30**. For example, the inventory data may be communicated to inventory visibility module **74** so that it may be organized and integrated into views that may be communicated to users associated with participants **30**. As described above, the data may be organized so that it may be presented to a user having a particular role and having permission to view particular data. Furthermore, inventory visibility module **74** may monitor the data or information generated from the data and determine when an exception has occurred and/or when an alert should be communicated to particular participants **30**. Business analytics module **76** may also be used to analyze the inventory data for auditing, tracking, trend analysis, or for other types of data analysis. Processing may also occur in association with one or more agents associated with DIMS **20** that are responsible for performing particular business functions with respect to the inventory data.

Alternatively or in addition to performing step **104**, DIMS **20** may communicate inventory data to one or more engines **50** at step **106**. Engines **50** process this inventory data at step **108** using suitable techniques for the particular engine **50** and type of data. For example, inventory data from multiple participants **30** may be communicated to replenishment planning engine **50***e* for generation of a replenishment plan and/or schedule, inventory data may be communicated to an order management system **50***a* for generation of appropriate orders to replenish depleted inventory indicated by the inventory data, and/or any other suitable processing may be performed on the received inventory data by these or any other suitable engines **50**. Engines **50** may update an associated plan or schedule whenever inventory data or set-up information is changed by a participant **30** or on a periodic basis (for example, by requesting particular inventory data stored by DIMS **20**). A generated plan may also identify exceptions that have occurred due to a change in inventory data. One or more modules of DIMS **20** may also or alternatively analyze a plan or schedule and identify any exceptions (and generate alerts for particular participants **30**, if appropriate). At step **110**, engines **50** communicate the planning and/or scheduling information to DIMS **20**.

At step **112**, DIMS **20** communicates inventory information generated by DIMS **20** and/or engines **50** to appropriate participants **30** (such as to systems **32** of particular participants **30**) using integration module **60**. The inventory information may be communicated based on a request by a participant **30** (such as by a system **32** of a participant **30**), based on the generation of data by an engine **50** and/or a module of DIMS **20** that is relevant to a participant **30**, based on an alert generated for a particular participants, based on a pre-planned schedule, and/or for any other appropriate reasons. Therefore, this communication of information may be on a "push" or a "pull" basis. For example, a seller **44** may request a view of inventory of a particular item at one or more locations and such a view may be communicated to the user. As another

example, a module of DIMS **20**, such as inventory visibility module **74**, may evaluate incoming inventory data from a supplier **40**, recognize that the inventory level of a particular item has dropped below a pre-defined level, and communicate an alert to the supplier **40** and an associated seller **44**. As yet another example, an engine **50** may identify an exception in a plan or schedule generated from inventory data received from DIMS **20** and DIMS **20** may communicate the existence of such an exception to appropriate participants **30**. In general, DIMS **20** serves as a collection point for inventory information for multiple participants **30** in a value chain and DIMS **20** may serve to communicate the "raw" inventory data of one participant **30** directly to other participants **30**, process the data internally and communicate the results of this processing to one or more participants **30**, and/or communicate inventory data to appropriate engines **50** for processing and communicate the results of this processing to appropriate participants **30**. At step **114**, participants **30** receive information communicated from DIMS **20** and take appropriate action.

Although the steps of the example method described above, as well as those described below, are illustrated as occurring in a particular order, it should be understood that the steps may occur in any appropriate order and different steps may occur simultaneously for different instances of inventory data or for other suitable reasons. Furthermore, one or more of the steps may not be performed and/or additional steps may be include as appropriate for the operation of DIMS **20**.

As mentioned above, intelligent fulfillment agents may be used by DIMS **20** to conduct various tasks. These fulfillment agents facilitate the triangulation of relevant participants **30** in the context of the various fulfillment and replenishment workflows that occur between the participants **30**. Unique business relationships and rules are captured in configurable fulfillment engagement rules that leverage the inventory visibility provided by DIMS **20**. The triangulation takes place at two levels: the participant relationship level and the data/information level. In the former, the participants **30** involved are modeled (for example, a buyer **48**, a seller **44**, and fulfillment/execution service provider **46**). The data/information level contains the data and information includes the inventory and other related information provided by DIMS **20**. The fulfillment agents enable a seamless flow of information between these two levels thus provide an event-based replenishment transaction management with rich decision support. The agents are also intelligent in that they can adapt to execution level exceptions changes that occur, such as at a planning or strategic level in the value chain. For example, if changes to the network set-up occur or customer demand increases occur, these agents can recognize such changes and use the information about the inventory positions in the value chain to react as necessary with appropriate workflows.

The flexible framework will allow participants **30** to adapt to changing market needs and business rules. Two fundamental enablers of this framework are visibility and connectivity. The visibility provides a view of the inventory both at a discrete level (for example, by SKU, location, or owner) as well as an aggregated view across a distributed environment (for example, a single seller **44** with multiple sites or multiple sellers **44**). The transaction management capability of DIMS **20** will let the users track inventory as it changes ownership or is being co-managed by the various entities involved in the fulfillment and replenishment workflows.

Different fulfillment agents may be used for different types of fulfillment and replenishment workflows. For example, one agent may be used for a JIT workflow and another may used for a VMI workflow. Although many other types of agents may be implemented through DIMS **20** to enable any suitable fulfillment and replenishment workflows, for purposes of example a JIT fulfillment agent will be described below in further detail.

There are four main aspects of a typical JIT implementation. The first such aspect is a contract(s) between the relevant participants **30**. For example, in a JIT environment, a supplier **40** delivers raw materials and other purchased items as they are needed. A blanket purchase order or other suitable form of basic agreement typically covers the terms and conditions for this procurement. Generally, the relevant participants agree on acceptance criteria before they embark on the JIT program. A set of key performance indicators (KPIs) are often selected to measure the effectiveness of the program and to set criteria for the participants **30** to follow. For example, KPIs may address the service level of the suppliers **40**, the inventory turnover, the fill rate, and/or any other suitable measures.

The next aspect of JIT is planning. JIT typically requires a flow of material in the exact quantity required at an exact time. Regardless of the specific method used to achieve this exact material flow, there must be advance planning to ensure that material is available when needed. A technique such as material requirement planning (MRP) is generally used to accomplish this task. In addition to predicting the future material requirements, just-in-time replenishment quantities and minimum/maximum levels need to be determined. Furthermore, early warning visibility and assistance in decision-making related to unexpected events is also needed.

A JIT implementation also involves electronic data interchange (EDI) message transmission and translation. In many value chains, value chain management extends to the sharing of planning, operational, tactical, and strategic data. This includes sharing short-term scheduling data, medium-term materials requirements planning and scheduling, and longer-term forecasts. The traditional technique for performing JIT communications has been the use of EDI over value added networks (VANs). The disadvantages of EDI-VAN are that the use of VANs is expensive, there are many trading partner-specific variations on EDI standards (and thus communication problems are created), and implementing EDI is a costly and technologically challenging process.

Configuration is also an aspect of JIT. Depending on the industry, there are many configurations possible. Some sellers **44** may want to own inventory and typically the suppliers **40** will deliver the material directly to a seller's location when needed. Another alternative may be that sellers **44** establish their own warehouses near a seller's facility and material is delivered in a just-in-time manner when required by a seller **44**. Some sellers **44** may opt to have a hub (typically a warehouse managed by third party service provider **42**) and suppliers **40** deliver the products directly to hub so that material can pulled on a just-in-time basis from the hub. In this case, either the supplier **40** or the third party service provider **42** owns the inventory depending on the business scenario.

JIT replenishments are normally focused on short-term order fulfillment. The following are example steps of a typical JIT fulfillment process. The seller **44** (such as a manufacturer) sends forecast information and a planning schedule to its suppliers **40** on a periodic basis (for example, weekly). Suppliers **40** use this information for their internal planning. These forecasts typically span a forward horizon of eight to twelve weeks. Seller **44** also sends a shipping schedule to suppliers **40** on a daily or weekly basis. Depending on the business practice, suppliers mayor may not be required to provide commits against the shipping schedule. The shipping schedule in a traditional MRP is a push communication. It basically tells the supplier **40** what items the seller **44** wants

the supplier **40** to ship. The seller **44** expects to use these items and it wants the supplier **40** to deliver the items whether they end up using the items or not.

On a daily basis, seller **44** monitors the inventory levels and demand signals and sends EDI change notifications to first tier suppliers **40** to adjust the shipping schedules. First tier suppliers **40** examine the contents of the schedules and determine whether or not they can meet the requested quantity. Each first tier supplier **40** then sends an acknowledgment to seller **44**. First tier suppliers **40** also input the shipping schedule information into their planning systems to use the information as the basis for their forecast demand. However, daily shipping schedules constantly vary from forecast requirements, so safety stocks (inventory buffers) are added into the calculations and net requirements for raw materials and subassemblies or components are calculated. These planned net requirements for procured inventory are then communicated weekly to second tier suppliers **40** (suppliers to the first tier suppliers), sometimes via EDI or more commonly via fax. However, as a result of daily changes in OEM actual demand, first tier suppliers **40** frequently issue daily changes to their planned requirements from their suppliers **40** (the second tier suppliers **40**). This same process is repeated between second tier suppliers **40** and third tier suppliers **40**, and so on. However, EDI is rarely the communication medium between these lower tier suppliers. More typically, the communication medium is fax, phone, or mail. The net result is considerable delay in communicating requirements to lower levels in the supply chain. As a result, suppliers **40** to the first tier suppliers **40** try to respond to planned requirements several weeks in advance and compensate by building up safety stock levels to accommodate the eventual wide variability in demand.

The first tier suppliers **40** each generate an advanced ship notice once a truck going to the seller **44** is loaded. This transaction notifies the seller of the actual contents of the truck and can be helpful for scheduling the receiving dock and identifying cross-docking opportunities. Similarly, shipping notices also are sent between suppliers **40** at the different tiers. Again, the communication mode may be EDI, but is more commonly fax, phone or email in these cases. The truck carrying the product typically arrives at the seller's **44** dock at its fixed appointment time and is promptly unloaded so that the product can be available to the line. The replenishment lead-time usually is quite short and consists of mainly the in-transit time, which can range from a few hours to few days.

Although the JIT process described above may have advantages over other types of processes, there are also many challenges associated with this traditional JIT fulfillment process. Such challenges include demand variation challenges. Customers (which may include sellers **44** and suppliers **40**) generally push variation in demand down to their suppliers **40**. This is slow, cumbersome, and overly expensive because forecast data is batched and depends on heavy administrative intervention even where EDI systems are in use. This process inherently builds excess inventory to accommodate wide variations in daily demand. An insidious side effect to the volatility in daily demand is that production schedules are difficult to plan and maintain. This creates inefficiency and causes excessive overtime to be worked. Although a seller **44** provides the forecast data to its suppliers **40**, the shipping schedule (JIT material requirements) almost always differs from the forecast because of the demand variation.

There are also communication challenges associated with the traditional JIT process. Although a seller **44** and a first tier supplier **40** may have good relationships and may be communicating through well-established EDI communication programs, this EDI communication gets expensive as more and more participants **30** join the JIT program. In many instances this leads to custom data mapping and maintaining these mappings can get very expensive. On the other hand, the level of sophistication decreases as you move down the supply chain and the downstream suppliers **40** rely mostly on fax, phone and e-mail messages for communication with upstream suppliers **40**. In order to compensate for the inefficiencies with such communication techniques, more and more inventory is added to support upstream demand. Furthermore, due to improper communication set-up, delivery problems from lower tier suppliers **40** are often not communicated to upper tier suppliers **40**, or such problems result in fragmented, duplicated, or delayed communications. This affects the supplier reliability and creates a lack of trust throughout the value chain.

The traditional JIT process also has visibility challenges. Since traditional JIT fulfillment is an independent process, there is not visibility into any of a seller's planning or transactional data. It also does not contribute to increasing the visibility of the seller's planning processes. Lack of visibility into multi-tier participant inventory (for example, a fourth tier supplier's lack of visibility into the inventory of a first tier supplier **40** or a seller **44**) poses a challenge to suppliers **40** since they have to deal with their customer's near term demand changes and end up building more "just-in-case" inventory to satisfy changing demand needs. Changes in order sequence, inventory buildups, delayed response from lower tier suppliers as well as transport delays could be avoided by providing global visibility to all appropriate participants **30**.

Because of the above challenges, first tier suppliers **40** spend too much time and money managing and directing their suppliers **40**. This approach is labor intensive, results in unreliable part deliveries, and generates high inventory levels in the supply chain. It also results in excessive expediting and high administrative costs. Furthermore, last minute demand and schedule changes result in high premium freights for the suppliers **40**. Suppliers **40** also have to consume the high inventory carrying costs that result from inventory they build to safeguard themselves from unpredictable variation in daily demand changes. All of this adds significant cost to the value chain.

The use of a DIMS fulfillment agent for JIT fulfillment can reduce or eliminate the above problems with the traditional JIT fulfillment process. JIT fulfillment agents and other fulfillment agents associated with DIMS **20** may be implemented as software associated with DIMS **20** or may be associated with a participant **30** an be programmed to use information provided through DIMS **20** to accomplish one or more specific business functions that are the responsibility of each agent. As described above, DIMS **20** provides visibility into a buyer's demand signals and planning forecasts for all relevant participants **30** in the value chain. JIT fulfillment agents associated with DIMS **20**, as well as the use of other types of agents provided through DIMS **20**, may serve to automate the process of monitoring relevant information from participants **30**, provide exception monitoring and generation of automatic alerts for participants **30** as and when particular events occur (such as alerts generated when inventory levels are violated), and initiate appropriate business processes in response to this information and events.

Specifically, JIT Fulfillment agents can be used to employ different business models ranging from single-tier model to multi-tier models. In both cases, participants **30** may have one-to-one (a single seller **44** interacting with a single supplier **40**), one-to-many (a single seller **44** interacting with multiple suppliers **40** and vice versa) and many-to-many

(multiple sellers **44** interacting with multiple suppliers **40**) relationships. The basic concept of JIT fulfillment remains the same whether it is a single tier or multi-tier value chain or whether one-to-one or one-to-many relationships are modeled. Although the concepts described herein primarily relate to the relationships between a seller **44** and a supplier **40**, these concepts are applicable between any two participants **30** in a value chain.

As described above, DIMS **20** provides an integration service that communicates with systems **32** to get the relevant inventory and related business data as and when required. All relevant data may be stored at a single location (for example, at a seller location **44**) or may be distributed at multiple locations (but integrated) and is made available to agents to automate businesses processes, such as a JIT process. Using such agents, a company can interconnect all of its trading partners and keep everyone synchronized with each other's current plans while tracking actual activity in real-time across companies. This increases the velocity in the value chain and results in increased efficiency, increased planning accuracy, improved ordering and inventory control, reduced inventories (and thus costs) throughout the supply chain.

FIG. **4** illustrates an example JIT fulfillment workflow that may be implemented and automated using JIT fulfillment agents associated with DIMS **20**. The method begins at step **200** where a JIT agent associated with or in communication with DIMS **20** initiates and triggers a request from DIMS **20** to a seller **44** (such as a manufacturer) for a planning schedule for long-term planning (such as an EDI **830** message) and for a product inventory report (such as an EDI **852** message) at a predefined time interval (for example, daily or weekly). At step **202**, the seller's inventory management system communicates the requested planning schedule and product inventory report to DIMS **20**. This information may be communicated to VMI **20** in response to the request or at an appropriate user-defined time interval. Integration module **60** allows the receipt of this information from the seller's inventory management system **32** and may convert the information (for example, in an EDI message or messages) to an appropriate format for processing. The JIT agent may direct this translation based on rules for data conversion for the particular JIT program.

At step **204**, DIMS **20** persists the scheduling data and updates the seller's stock levels for all SKUs based on the product activity data. At the direction of the JIT agent, DIMS **20** communicates the seller's forecast update based on the schedule to the suppliers' inventory management system **32** at step **206** (the communication may be translated if appropriate for each supplier **40**). Alternatively, an agent may exist at the supplier **40** and have rules for data and format conversion for that particular supplier **40**. At step **208**, the relevant suppliers **40** use this data to plan their forecast and, in turn, communicate their forecast to downstream suppliers **40** using the JIT agent and DIMS **20**, as described above.

At step **210**, the JIT agent monitors the seller's stock levels communicated to DIMS **20** and sends an alert to the supplier at step **212** based on a JIT reorder strategy implemented as rules associated with the agent. Such an alert may communicated to a system **32** or directly to a user via e-mail, page, fax, or using any other appropriate communication technique. In addition or alternatively, the agent can create a shipping or delivery schedule (such as an EDI **862** message) and communicate this schedule at step **214** to relevant suppliers **40** when the inventory level reaches a pre-set just-in-time reordering point. Such a schedule may be communicated to the seller **44** for confirmation and seller **44** may modify it and upload the modified schedule to DIMS **20** at step **216**. Furthermore, at step **218** the agent may communicate relevant information to order management system **50*a*** for the creation of a purchase order for the items that are needed to increase the stock level. At step **220**, order management system **50*a*** communicates one or more appropriate orders to DIMS **20**, and agent directs the communication of these orders to appropriate suppliers **40** at step **222**.

At step **224**, the relevant suppliers **40** communicate an acknowledgement of the alert, shipping schedule, and/or purchase orders. If the supplier cannot meet the shipping schedule, an exception workflow may be triggered. At step **226**, the suppliers **40** perform transportation planning for the orders and communicate an order status to DIMS **20**. The JIT agent directs the communication of the order status to seller **44** at step **228** to update the seller's order management system. Similarly subsequent order statuses like advanced shipment notices from suppliers **40** may be updated for an order in DIMS **20** and relayed to the seller **44**. At step **230**, the JIT agent also requests and receives the order status from relevant service providers **42** during transportation to update both the seller **44** and supplier **40**. The JIT agent tracks the order until the status of the order is considered closed by both the participants **30**, and the example method ends. All of the actions of the JIT agent described above may be the result of rules associated with the agent that dictate the action the agent takes when a particular information is received from a participant **30** (such as through a system **32**), from an engine **50**, and/or from any other appropriate source.

Although a JIT fulfillment agent is described in the example method above, numerous other types of agents may be used in conjunction with DIMS **20** to implement rules to execute a particular business function. For example, agents can be programmed to execute other replenishment programs, such as VMI or SMI, in a manner similar to that described above. Furthermore, agents can be programmed to evaluate data received from participants **30** and/or engines **50** and to generate alerts based on rules that are used to identify issues with the data (for example, maximum and/or minimum levels of a particular data measure, violation of timing-based tolerances, or mismatched in ordered and delivered items). Agents may also generate alerts based on the absence or presence of a transaction (for example, the non-receipt of an expected EDI message). Agents may also be used to perform and/or initiate a business process, such as the creation and communication of a purchase order using an order management system based on an evaluation of inventory data from one or more participants **30**. As described above, agents may also automate a series of tasks, such as automating the numerous tasks involved with a replenishment program (such as the JIT program described above). Agents may further be used to automate any other appropriate tasks associated with inventory management and thus may be used to efficiently leverage the inventory visibility provided by DIMS **20**.

In addition to the advantages described above, the connectivity and visibility provided by DIMS **20** may also be used to perform distributed order fulfillment. Order fulfillment is the process of managing a customer order through its entire life cycle. Broadly classified, the life cycle of the order spans the processes of capture and creation, verification and approval, quoting, sourcing, planning and scheduling, execution, and returns management. The specifics within these order processes often vary to a high degree between industry segments and also between different sellers in the same industry segment. However, these different workflows have some basic similarities that can be traced to the business models followed by these sellers. These business models and their influence on existing order fulfillment processes are described below as

context to the description of the novel distributed order fulfillment process made available using DIMS **20**.

One aspect of order fulfillment is intra-enterprise fulfillment between the different portions of an enterprise. Intra-enterprise order fulfillment may be a challenge for sellers **44** having multiple divisions each having a separate order fulfillment process. These divisions typically may have disparate systems managing their order fulfillment life cycle and there has often been little visibility and coordination between these divisions. Similarly, a seller **44** may have multiple stocking locations for a product and ship from any of these locations depending on the buyer **48** and destination of an order. However, once a sourcing location has been identified and the order information has been sent to this location, the seller **44** may loose visibility to the order until it is actually delivered to the buyer **48**. Having visibility to an order during order planning and execution steps and managing any exception thus becomes an expensive and time-consuming process for the seller **44**.

For example, many large sellers **44** typically have multiple divisions that are managed and operated independently. These divisions may be organized by the product lines or by regions or by industry segments. Many of these divisions have different operating practices and business processes in place. They may also have disparate seller applications managing these business processes. As a result, it is difficult to get an aggregated view of data associated with the seller **44** across all these divisions. Because of the lack of an integrated environment, such sellers **44** often direct a buyer **48** to place an order with the division concerned and buyer **48** ends up sending multiple orders to different divisions of the same seller **44**. This results in an increased ordering cost for the buyer **48** and increased order maintenance cost for seller **44**. Some sellers **44** do use a single Internet store-front that allow a buyer **48** to place an order and then internally direct the order to the right division, but such a system still lacks the visibility at the back end of the supply chain where the actual order fulfillment takes place. This makes the order coordination across divisions a difficult task for a seller **44**.

Furthermore, it is typical for a seller **44** to have multiple stocking locations from where a product can be shipped to a customer. The stocking locations may be distribution centers in different regions and/or warehouses near the seller's manufacturing plants. Depending on the number of these locations and the products stored each location, the sourcing decision for a customer order can become complex to manage effectively. Most of these sourcing decisions are static in nature, such as a distribution center in a region supplying to the customers in that region to minimize the transportation cost. These static rules often lead to situations where a customer order is delayed in one distribution center because of product shortage even though a distribution center in a different region may have excess of the product.

Problems also exist with order fulfillment in an multi-enterprise ordering environment. In such an environment, buyers **48** depend on a multitude of sellers **44** for various products (or sellers **44** depend on a multitude of suppliers **40**). Because these sellers **44** are different entities, the buyers **48** work with these sellers **44** independently to manage the order fulfillment process. For example, a buyer **48** would typically have a contract in place with a number of sellers **44** of different products. During an order fulfillment process, the buyer **48** would place purchase orders separately to these sellers **44** for the specific products that they supply. The buyer **48** would then follow-up independently with each of these sellers **44**

during the order life cycle. This leads to increased ordering cost and makes the order coordination between sellers **44** extremely difficult.

Most of the order fulfillment challenges for participants **30** in a value chain result from the lack of visibility into each other's supply chain and lack of information sharing between participants **30**. Buyers **48** often do not get the right promise from their sellers **44** on their orders and thus do not get the product at the right time. This forces them to maintain a higher level of inventory for the critical products or components, which leads to higher carrying cost. Similarly a buyer **48** often cannot place a consolidated order to multiple sellers **44** and be able to track the order through its whole life cycle. This results in higher ordering cost and order management cost.

Many of sellers' fulfillment-related problems stem for an inaccurate demand information from a buyer **48**. Even when the demand planning is within acceptable limits, the sellers **44** still incur major fulfillment-related problems because their material, product, and/or capacity profiles are unreliable. The most common causes of this include internal production or procurement problems, in addition to poor communications with the buyers **48** (often due the high connectivity costs). These problems contribute to excessive fulfillment cycle time, expediting costs, poor margins and service levels and, ultimately, lost business.

DIMS **20** may be used to provide a distributed order fulfillment service that provides participants **30** with the infrastructure and a configurable set of workflows to manage the entire life cycle of a distributed order (for example, a multi-seller or multi-divisional order). The distributed order fulfillment process implemented using DIMS **20** may be used to manage the entire life cycle of an order starting with order capture and promise and ending with a proof of delivery and subsequent financial settlement of the order. To provide these functions, appropriate order management and fulfillment modules may be added to or associated with DIMS **20** in particular cases or such functions may be implemented, in part, through appropriate engines **50** in communication with DIMS **20**. This distributed order fulfillment process provides fast and accurate order promising, real-time order status reporting for a seller **44** and for its buyers **48**, dynamic sourcing decision-making for optimal order fulfillment, and efficient exception management and error handling.

FIGS. **5**A and **5**B illustrate an example distributed order fulfillment method. As described above, order fulfillment workflows may vary greatly between the industry segments and also between different sellers **44**. Therefore, an example generic distributed order fulfillment process is described below. This example process may be divided into two segments: order fulfillment and order execution. Order fulfillment includes the processes of capture and creation, verification and approval, quoting, sourcing, and scheduling of an order. Order execution includes the processes of warehouse planning and execution and transportation execution and management.

The method begins at step **300** where a buyer **48** creates an order for an item, such as a product. Although a buyer **48** is described as purchasing items from sellers **44**, it should be understood that the process applies equally to sellers **44** purchasing items from suppliers **40**, suppliers **40** purchasing items from other suppliers, and any other appropriate transactions between participants **30** in a value chain. Some of the parameters included in the created order may be the requested item(s), the corresponding due date and quantity of each item, and the desired sellers **44** for an item (for example, the buyer **48** may have existing contracts with particular sellers **44**).

Buyer **48** may alternatively specify a list of sellers **44** who are eligible to supply an item in the order.

Once the order is created, buyer **48** communicates the order to DIMS **20** at step **302** for either a quote (using a request for quote) or a promise (using a request for promise). For example, if buyer **48** is EDI-enabled, buyer **48** may send an EDI **840** message to DIMS **20** for a quote or an EDI **850** message to place a confirm order. For the purposes of this description, it is assumed that buyer **48** first communicates a request for quote. At step **304**, DIMS **20** may send a functional acknowledgement (such as an EDI **997** message) to buyer **48** upon receipt of the request. At step **306**, DIMS **20** determines, using the information about participants modeled in DIMS **20**, which sellers **44** should receive the request based on the items requested in the order and, if applicable, the set of potential suppliers requested by buyer **48**. Depending on the number of items included in an order and the sellers **44** requested by buyer **48**, each seller **44** might receive one or more line items for a quote. This function, as well as the other functions of DIMS **20** described below, may be performed using one or more of the modules described above and/or using one or more additional modules (such as a fulfillment server) added specifically for distributed order fulfillment functions. Again, the various modules of DIMS **20** may be co-located at one location (for example, at a location associated with a participant **30** or a third party) or they may be located at multiple locations (and potentially replicated at multiple locations). For example, relevant sellers **40** may be identified using data from relationship modeling module **64**. If the terms of the buyer's contracts with one or more sellers **44** are communicated to and stored by DIMS **20**, DIMS **20** may access these contractual terms to determine appropriate sellers **48** for the buyer's requested items and the corresponding prices for each item, if specified in the terms.

At step **308**, DIMS **20** communicates the buyer's request to the appropriate sellers **44** (if a seller **44** is EDI enabled, DIMS **20** may communicate an EDI **840** message to the seller **44**). At step **310**, the selected sellers **44** receive the request for quote and generate a response to the quote. The generation of the response is an intra-seller task and the mechanism for generating such a quote may range from a completely manual process to the use of a sophisticated order-quoting engine, such as a demand fulfillment engine.

At step **312**, DIMS **20** may receive the responses from the sellers **44** (possibly over a window of time specified by buyer **48** during which responses will be accepted). DIMS **20** consolidates these responses into a single response to the original request at step **314**. The business logic of consolidation of the individual responses is configurable by buyer **48**. For example, one buyer may want to see all the responses from all the sellers **44** and then select the best response, whereas another buyer may prefer that DIMS **20** determine, based on pre-configured business rules, the best response to the request. These business rules may be implemented in an agent or using any appropriate modules of DIMS **20**. At step **316**, DIMS **20** communicates the consolidated response to buyer **48**. If buyer **48** is EDI-enabled, an EDI **843** message may be sent to buyer **48**.

Upon receiving the response, buyer decides at step **318** whether to reject the quote or place a confirm order against the quote. If buyer **48** decides to place a confirm order, buyer **48** communicates a purchase order to DIMS **20** at step **320** (if buyer **48** is EDI-enabled, buyer **48** may communicate an EDI **850** message for order confirmation). It is assumed that the purchase order creation is an intra-enterprise activity for buyer **48**. The mechanism used by buyer **48** to create a purchase may include a manual process, an ERP system, an order management system, or any other appropriate mechanism. At step **322**, DIMS **20** receives the purchase order and may validate the order according to business rules established between buyer **48** and a seller or seller **44**. At step **324**, DIMS communicates an acknowledgement of the receipt of the purchase order (such as an EDI **997** message).

Once the order is received by DIMS **20**, it is persisted in DIMS **20** or an associated engine **50** (such as an order management engine **50***a*) and is given an appropriate order status at step **326**. At step **328**, if the order is based on a previous quote, DIMS **20** assigns a promise to the order based on both earlier responses by the sellers **44** and/or the buyer's selection of a quote and the order status is adjusted appropriately to indicate the promised state. In addition or alternatively, DIMS **20** may provide a real-time promise to buyer **48** on behalf of a seller **44** based on inventory information provided to DIMS **20** by the seller **44**. Therefore, DIMS **20** provides buyers **48** with a selective view to the sellers' inventory to help plan their own orders. If a seller **44** is not used for final confirmed purchase order, DIMS **20** may notify the seller **44** with an appropriate message at step **330**. At step **332**, DIMS **20** communicates a message to the buyer (such as an EDI **855** message) for purchase order acknowledgement.

At step **334**, DIMS **20** determines appropriate destinations to which the order is to be sent and communicates the order to the appropriate destinations associated with sellers **44** at step **336**. Each seller **44** receives a purchase order that only consists of the line items concerning that seller **44**. If the supplier is EDI-enabled, DIMS **30** may communicate an EDI **850** message to seller **44**. As described above, a seller **44** may have multiple independent divisions or multiple warehouses supplying a specific item. In addition, the seller **44** may have multiple suppliers **40** that can either supply the item to the seller **44** or drop-ship the item to the buyer **48** for the seller **44** if needed. In scenarios like this, the seller **44** has previously needed to search through all of the divisions and/or warehouses to find out the item availability and the sourcing location. The sourcing logic can become extremely complex and can become unmanageable for a large seller **44**. In addition, once an item it is sourced to a division or warehouse, the seller **44** may loose visibility to the order at the seller level because of the lack of visibility to the divisional operation. However, DIMS **20** provides sellers **44** with a scalable and configurable framework to implement these complex sourcing rules and automate the sourcing process.

For example, assume that a buyer **48** requests that an item be shipped to its New Jersey distribution center from a seller **44** that stocks the item in distribution centers based in Los Angeles and Chicago. If minimizing the transportation costs is one of the sourcing criteria, then the business rules can be configured in DIMS **20** to source the item from Chicago instead of Los Angeles (assuming it will cost less to ship from Chicago). As another example, DIMS **20** may source the order from a the warehouse of a seller **44** having the most unpromised inventory of the ordered item (according to the inventory data provided to DIMS **20**). DIMS **20** provides sellers **44** a scalable framework to conveniently model the various sourcing rules that may be used. Once the sourcing logic is configured in DIMS **20** for a seller **44**, then the purchase order from buyer **48** can be communicated directly to the particular division or warehouse selected at step **336**.

At step **338**, each seller **44** (as described above, the term seller as used here may refer to a particular division of a seller **44** or other destination associated with a seller **44**, and each destination may be modeled in DIMS **20** as a participant **30**) receives the appropriate order and may compare the original promise made in the response to the request for quote (made

by a seller **44** and/or made by DIMS **20** on behalf of seller **44**) to the latest promise based on the latest supply position. If the promise is different, then the seller **44** may initiate a change promise workflow at step **342** and notify the buyer **48** via DIMS **20** (in cases when the purchase order is not a result of a quote process, seller **44** sends the promise information to DIMS **20**). If the seller **44** is EDI-enabled, it may send an EDI **855** message (purchase order acknowledgement) to DIMS **20** to update the earlier promise or send a new promise. At step **344**, DIMS **20** updates the promise for the seller **44**, if needed, and notifies buyer **48** regarding any change (for example, by communicating an EDI **855** message to buyer **48**). This completes the order fulfillment process for the order. It should be noted that while a seller **44** is responsible for only one or more line items of the buyer's purchase order, DIMS **20** keeps track of the buyer's complete purchase order and provides buyer **48** with real-time order status updates, as described below.

FIGS. **6**A and **6**B illustrate an example distributed order execution method. Order execution includes the processes of warehouse planning and execution and transportation execution and management. The order execution process typically involves tasks performed by the seller **44** and associated third party service providers **46** to deliver the order to the buyer destination (although, as described above, the process applies equally to sellers **44** purchasing items from suppliers **40** and using service providers **42**, suppliers **40** purchasing items from other suppliers **40** and using service providers if appropriate, and any other suitable transactions between participants **30** in a value chain). Although these tasks are mainly intra-enterprise to the seller **44**, the status of the order is keep updated and received by buyer **48** and other appropriate participants **30** using DIMS **20**. Furthermore, the various communications between the participants **30** described below may be communicated directly between the participants **30** and/or via DIMS **20**.

An order is often shipped by a seller **44** in shipments or loads. Grouping of orders into a load or shipment allows the seller **44** to minimize its transportation cost. Transportation planning is the process where the seller **44** groups the orders into optimized shipments or loads based on a set of criteria such as destination, source, routes, transportation cost, special handling instruction, and any other appropriate criteria. The transportation planning process may also take into account any merge-in-transit and/or value added service constraints on the order.

The example method begins at step **400** where the seller **44** carries out appropriate transportation planning for an order by grouping the order into loads or shipments. The outcome of such a planning process is a set of loads that consist of one or more orders with one or more items. Once the loads are created, the loads are tendered to one or more carriers **46** for transportation at step **402**. Load tendering is the process where carriers **46** are contacted with the load information (for example, source, destination, item, and pick-up/delivery window). The messages associated with transportation planning, as well as all other messages associated with order execution, may be communicated to the appropriate entity using DIMS **20** (and the message may be communicated directly to the entity, if appropriate). At step **404**, DIMS **20** receives the load tendering message and updates the status of the associated order (for example, the order status may change to "load tendered" status).

At step **406**, DIMS **20** communicates the load tendering message to appropriate carriers **46** and carriers **46** respond to the request. Seller **44** receives the responses (for example, using DIMS **20**) and selects a carrier **46** at step **408**. Seller **44** may have a predefined contract with a specific carrier **46** for

a set of routes or a region. If seller **44** and a carrier **46** are connected via EDI, the seller **44** may communicate an EDI **204** to the carrier **46** and carrier **46** may send an acceptance using an EDI **990** message (these messages may be sent directly or via DIMS **20**). At step **410**, DIMS **20** receives these messages and updates the status of the order based on the messages (for example, the order status may change "tender accepted" status).

Once a tender has been accepted by a carrier **46**, the loads are released at step **412** to an appropriate warehouse for warehouse order planning. The warehouse is also notified about the appointment time when the carrier's trailer should arrive at the warehouse to pick up the load. Seller **44** may send an EDI **940** message to a warehouse as the order release signal. DIMS **20** receives the order release communication at step **414** and updates the order status appropriately. Once a warehouse receives a release notice, the items related to the orders are identified, picked and packed at step **416**. This operation is usually a batch process and is done a few times during a typical day depending on the order volume. Any discrepancy that would cause an order to be shipped late is identified at step **418** and a message indicating any discrepancy is communicated to DIMS **20** at step **420**. At step **422**, DIMS **20** updates the order status appropriately and notifies the buyer about the exception. The buyer may then initiate a change order workflow, if desired.

Once the shipments are created, they are moved to the specified area or dock where the carrier is to load the order. At step **424**, the warehouse communicates an advanced ship notice (ASN) to DIMS **20** (for example, an EDI **856** message may be sent as an ASN). A typical ASN includes the order number, the items being shipped, the delivery date, the delivery window, and any other appropriate parameters. At step **426**, DIMS **20** receives the ASN and updates the order status. DIMS **20** communicates the update order status to buyer **48** at step **428**.

If a carrier **46** picks up a loads from the warehouse dock at the specified time, a confirmation message (such as an EDI **214** message) is communicated to DIMS **20** at step **430**. DIMS **20** receives this message and updates the order status appropriately at step **432**. If the carrier does not show up at the appointed time, the orders need to be re-planned for a future date and time. In such a case, an appropriate message is communicated at step **434** from the warehouse to DIMS **20** to identify this exception. DIMS **20** notifies the buyer of this exception through an exception management workflow at step **436**.

Once carrier **46** has picked up a load, carrier **46** may send a daily or other periodic en route shipment status updates to DIMS **20** at step **438**. At step **440**, DIMS **20** receives such updates and updates the status of the order. At step **442**, DIMS **20** may communicate the updated order status to buyer **48** as updated, on a request basis, or at any other appropriate times (in some cases, DIMS **20** may only communicate an update if an exception has occurred). Once the shipment is delivered to the buyer destination, the carrier receives a proof of delivery (POD) and communicates the POD (for example, and EDI **214** message) to DIMS **20** at step **444** to indicate the completion of the delivery process. An EDI **214** message may be sent for a POD confirmation. DIMS **20** receives the POD and updates the order status to indicate a POD status at step **446**. At step **448**, the updated order status is communicated to seller **44** (and may be directed to the specific warehouse which shipped the order, if appropriate).

The buyer at this stage can examine the contents of the shipment and communicate a message at step **450** identifying any discrepancy between the ASN and the actual delivery.

DIMS **20** communicates any such message to seller **44** at step **452**. Furthermore, DIMS **20** may provides an exception management workflow to handle these exceptions. At step **454**, buyer **48** and seller **44** complete a financial settlement workflow in which the supplier communicates an invoice (for example, an EDI **810** message) to buyer **48** and buyer **48** makes a payment to seller **44** (for example, using an EDI **820** message), and the example method ends. In particular embodiments, these financial settlement communication messages may be communicated between buyer **48** and seller **44** using DIMS **20**.

Although the present invention has been described with several embodiments, numerous changes, substitutions, variations, alterations, and modifications may be suggested to one skilled in the art, and it is intended that the invention encompass all such changes, substitutions, variations, alterations, and modifications as fall within the spirit and scope of the appended claims.

What is claimed is:

1. A distributed inventory management system, comprising: one or more computers coupled with a plurality of participant computers in one or more tiers of a value chain, the one or more computers configured to:

evaluate inventory data received from one or more of the plurality of participant computers according to one or more business rules;

execute a business process using one or more software agents, based on the evaluation of the received inventory data;

communicate output of the business process to one or more of the plurality of participant computers;

generate an alert when at least one of the plurality of participant computers uses a first replenishment policy in a first tier of the one or more tiers of the value chain that is different than a second replenishment policy used by at least one of the plurality of participant computers in at least a second tier of the one or more tiers of the value chain; and

communicate the generated alert to one or more of the plurality of participant computers.

2. The system of claim **1**, wherein the one or more business rules comprise a rule identifying maximum or minimum allowed levels of a particular inventory data measure.

3. The system of claim **1**, wherein the one or more business rules comprise a rule identifying a maximum time allowed for a fulfillment-related activity to occur.

4. The system of claim **1**, wherein the one or more business rules comprise a rule requiring a match between an ordered item and a delivered item.

5. The system of claim **1**, wherein the one or more computers is further configured to generate an alert based on the violation of a rule and communicate the alert to one or more of the plurality of participant computers.

6. The system of claim **1**, wherein the one or more computers is further configured to initiate a business workflow at a planning or execution engine coupled with the distributed inventory management system and communicate output of the engine to one or more of the plurality of participant computers.

7. The system of claim **6**, wherein the business workflow comprises a purchase order generation workflow in response to the violation of a rule specifying a minimum level of a particular inventory data measure.

8. The system of claim **1**, wherein executing a business process comprises initiating a replenishment program.

9. The system of claim **1**, wherein the one or more computers is further configured to model relationships between two or more of the plurality of participant computers specifying the identity of one or more of the plurality of participant computers, the locations of the one or more of the plurality of participant computers, the roles of the one or more of the plurality of participant computers of the value chain, and one or more business rules associated with the one or more of the plurality of participant computers.

10. The system of claim **1**, wherein the one or more computers is further configured to model one or more items based on the inventory data relating to the one or more items, the items comprises specifying the identity of one or more items relevant of the value chain, a description of the items, the locations at which each item is active, and an item hierarchy.

11. The system of claim **1**, wherein the one or more computers is further configured to:

communicate the received inventory data to one or more planning or execution engines coupled with the distributed inventory management system;

receive inventory planning or execution information from the engines; and

communicate the inventory planning or execution information to one or more of the plurality of participant computers.

12. The system of claim **11**, wherein one or more of the engines comprise a replenishment planning engine, an order management system, a demand fulfillment engine, a supply chain planner, or a factory planner.

13. The system of claim **11**, wherein the one or more computers is further configured to evaluate the inventory planning or execution information.

14. A computer-implemented method for distributed inventory management, comprising:

evaluating, by a computer, inventory data received from one or more of the plurality of participant computers according to one or more business rules;

executing, by the computer, a business process using one or more software agents, based on the evaluation of the received inventory data;

communicating, by the computer, output of the business process to one or more of the plurality of the participant computers;

generating, by the computer, an alert when at least one of the plurality of participant computers uses a first replenishment policy in a first tier of the one or more tiers of the value chain that is different than a second replenishment policy used by at least one of the plurality of participant computers in at least a second tier of the one or more tiers of the value chain; and

communicating, by the computer, the generated alert to one or more of the plurality of participant computers.

15. The method of claim **14**, further comprising:

initiating a business workflow at a planning or execution engine coupled with the distributed inventory management system; and

communicating output of the engine to one or more of the plurality of participant computers.

16. The method of claim **15**, wherein the business workflow comprises a purchase order generation workflow in response to the violation of a rule specifying a minimum level of a particular inventory data measure.

17. The method of claim **14**, further comprising:

communicating the received inventory data to one or more planning or execution engines coupled with the distributed inventory management system;

receiving inventory planning or execution information from the engines; and

communicating the inventory planning or execution information to one or more of the plurality of participant computers.

18. The method of claim **17**, wherein one or more of the engines comprise a replenishment planning engine, an order management system, a demand fulfillment engine, a supply chain planner, or a factory planner.

19. The method of claim **17**, further comprising evaluating the inventory planning or execution information.

20. Software for distributed inventory management, the software embodied in a computer-readable medium and when executed using one or more computers is configured to:

evaluate inventory data received from one or more of the plurality of participant computers according to one or more business rules;

execute a business process using one or more software agents, based on the evaluation of the received inventory data;

communicate output of the business process of one or more of the plurality of participant computers;

generate an alert when at least one of the plurality of participant computers uses a first replenishment policy in a first tier of the one or more tiers of the value chain that is different than a second replenishment policy used by at least one of the plurality of participant computers in at least a second tier of the one or more tiers of the value chain; and

communicate the generated alert to one or more of the plurality of participant computers.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,788,145 B2
APPLICATION NO. : 11/877087
DATED : August 31, 2010
INVENTOR(S) : Ganesh Wadawadigi et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Title Page, Item (75)

Replace Deepak Mohapatra with

Deepak Mohapatra, Richardson, TX (US)

Signed and Sealed this

Nineteenth Day of October, 2010

David J. Kappos

David J. Kappos
Director of the United States Patent and Trademark Office

EXHIBIT 5

US008781868B2

(12) **United States Patent**
Dogan et al.

(10) **Patent No.: US 8,781,868 B2**
(45) **Date of Patent: Jul. 15, 2014**

(54) **DETERMINING AN INVENTORY TARGET FOR A NODE OF A SUPPLY CHAIN**

(71) Applicant: **JDA Software Group, Inc.**, Scottsdale, AZ (US)

(72) Inventors: **Koray Dogan**, Boston, MA (US); **Adeel Najmi**, Plano, TX (US); **Mehdi Sheikhzadeh**, Irving, TX (US); **Ramesh Raman**, San Carlos, CA (US)

(73) Assignee: **JDA Software Group, Inc.**, Scottsdale, AZ (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **13/902,893**

(22) Filed: **May 27, 2013**

(65) **Prior Publication Data**

US 2013/0262177 A1 Oct. 3, 2013

Related U.S. Application Data

(63) Continuation of application No. 13/163,687, filed on Jun. 18, 2011, now Pat. No. 8,452,627, which is a continuation of application No. 10/836,448, filed on Apr. 29, 2004, now Pat. No. 7,966,211.

(60) Provisional application No. 60/470,068, filed on May 12, 2003.

(51) **Int. Cl.**
G06Q 10/00 (2012.01)

(52) **U.S. Cl.**
USPC **705/7.11**; 705/7.31; 705/7.25; 705/7.37

(58) **Field of Classification Search**
USPC 705/7.31, 7.25, 7.11, 7.37
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,946,662	A *	8/1999	Ettl et al.	....	705/7.26
5,953,707	A	9/1999	Huang et al.		
6,078,900	A	6/2000	Ettl et al.		
6,151,582	A *	11/2000	Huang et al.	....	705/7.25
6,643,556	B1	11/2003	Morenz et al.		
7,324,966	B2 *	1/2008	Scheer	....	705/28
7,584,116	B2 *	9/2009	Kakouros et al.	....	705/7.31

FOREIGN PATENT DOCUMENTS

TW	476889	2/2002
TW	495690	7/2002

OTHER PUBLICATIONS

Keaton, Mark: "Using the Gamma Distribution to Model Demand When Lead Time is Random", Journal of Business Logistics, vol. 16, No. 1, 1995.*

Keaton, Mark: "Using the Gamma Distribution to Model Demand When Lead Time is Random". Journal of Business Logistics, vol. 16, No. 1, 1995.

(Continued)

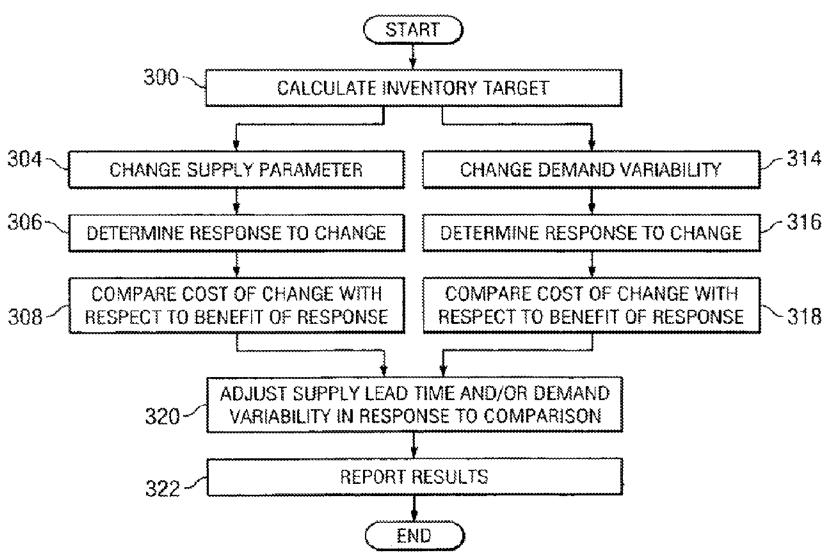
Primary Examiner — Mark A Fleischer

(74) *Attorney, Agent, or Firm* — Jackson White, PC; Steven J. Laureanti

(57) **ABSTRACT**

Determining an inventory target for a node of a supply chain includes calculating a demand stock for satisfying a demand over supply lead time at the node of the supply chain, and calculating a demand variability stock for satisfying a demand variability of the demand over supply lead time at the node. A demand bias of the demand at the node is established. An inventory target for the node is determined based on the demand stock and the demand variability stock in accordance with the demand bias.

20 Claims, 3 Drawing Sheets



(56) **References Cited**

OTHER PUBLICATIONS

Smith, Stephen A., "Optimal Inventories for an (S-I, S) System With No Backorders" Management Science, vol. 23, No. 5, Copyright 1977, The Institute of Management Sciences, pp. 522-528.
Wolff, Ronald W., "Poisson Arrivals See Time Avengers", Operations Research, vol. 30, No. 2, 1982 Operations Research Society of America, pp. 223-231.
Federgruen, Awi, et al., "An Efficient Algorithm for Computing Optimal (s, S) Policies", Operations Research, vol. 32, No. 6, 1984 Operations Research Society of America, pp. 1268-1285.
Svoronos, Antony, et al., "Evaluation of One-For-One Replenishment Policies for Multiechelon Inventory Systems" Management Science, vol. 37, No. 1, Copyright 1991, The Institute of Management Sciences, pp. 68-83.
"Improving Service and Market Share with Inventory Optimization; How to Improve both your top and bottom lines through superior inventory management", White Paper, i2 Technologies, Inc., Copyright 2003 i2 Technologies, Inc., 32 Pages.
Johansen, Soren Glud, "Base-stock policies for the lost sales inventory system with Poisson demand and Erlangian lead times", Department of Operations Research, University of Aahus, Denmark, pp. 1-14, Nov. 19, 2003.

* cited by examiner

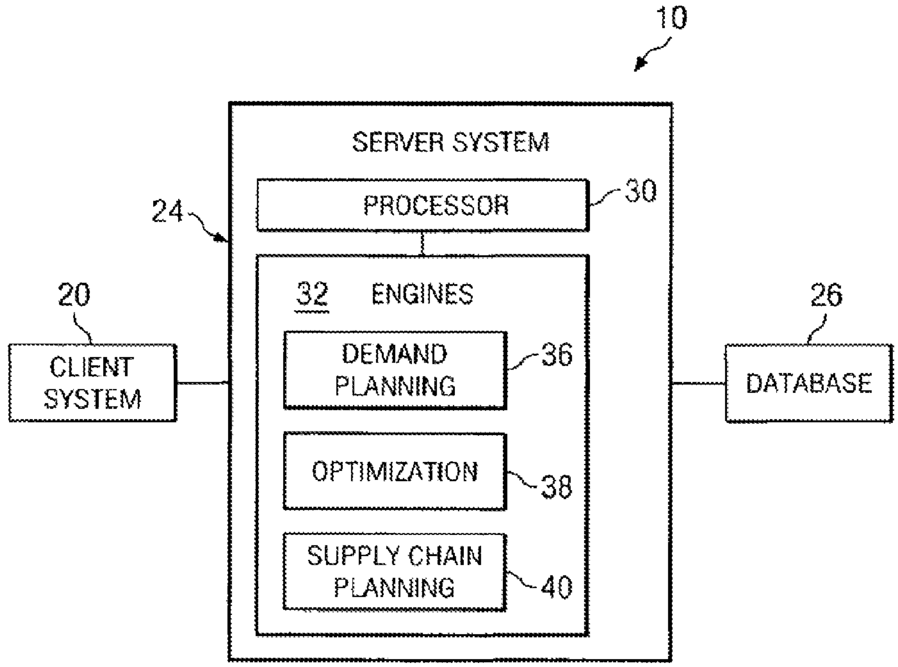


FIG. 1

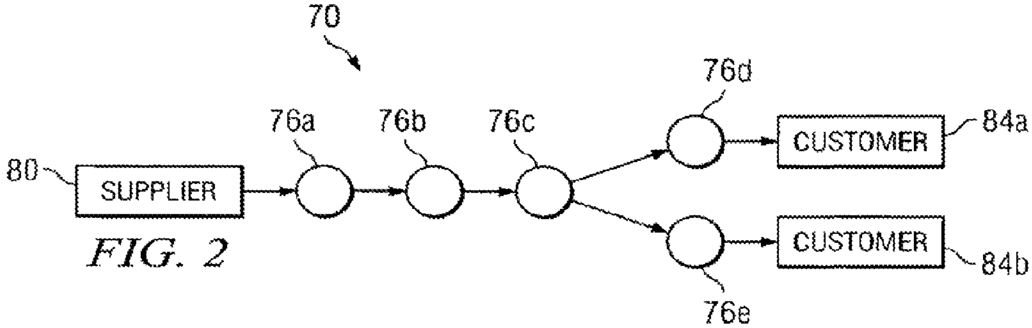


FIG. 2

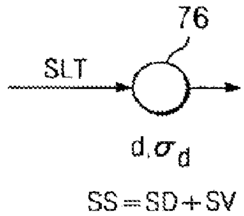


FIG. 3

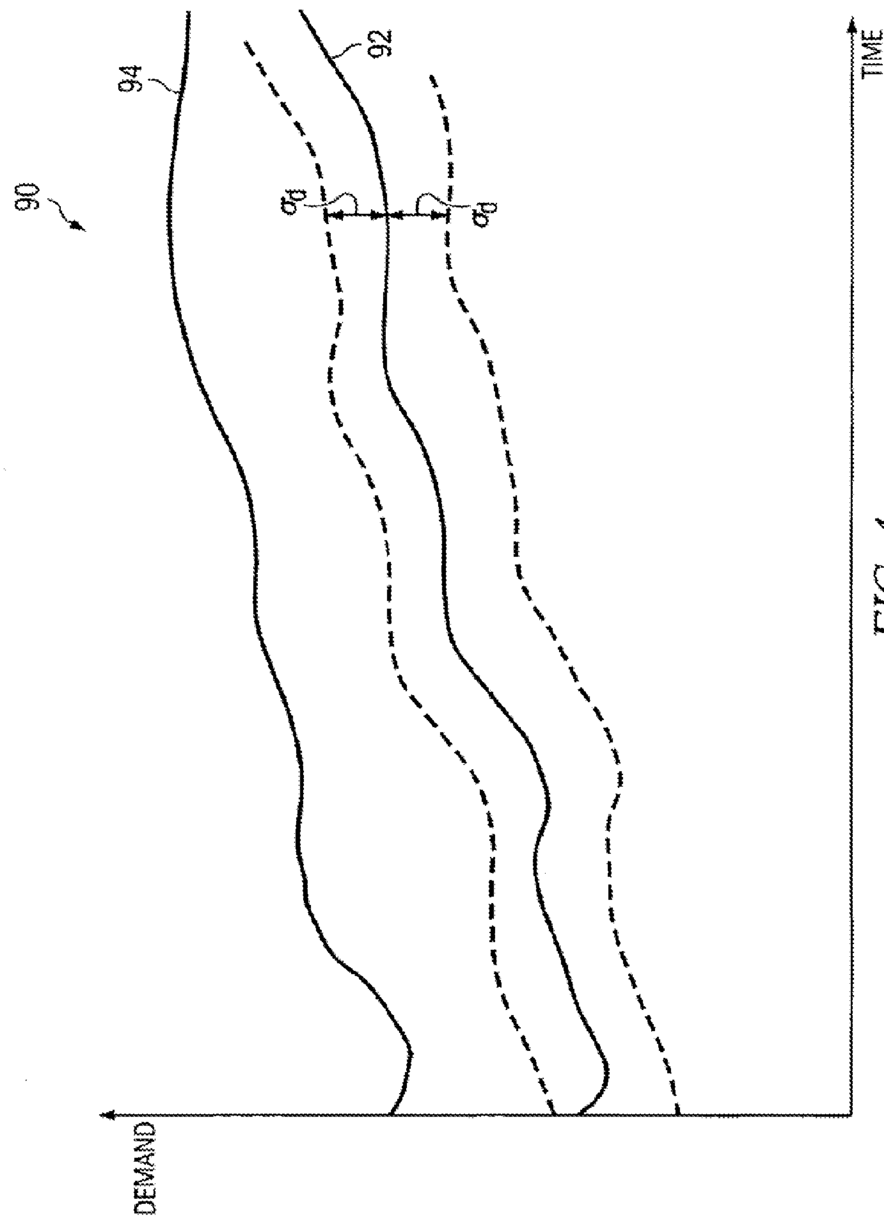


FIG. 4

FIG. 5

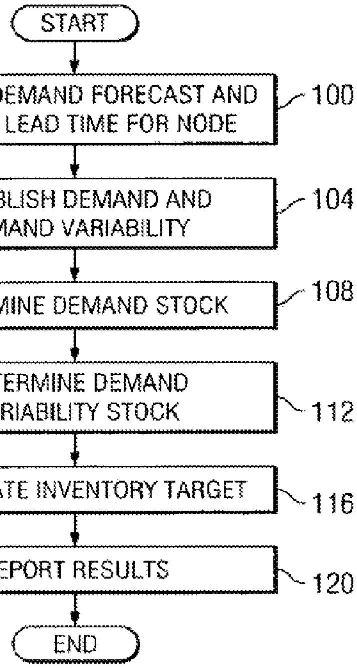
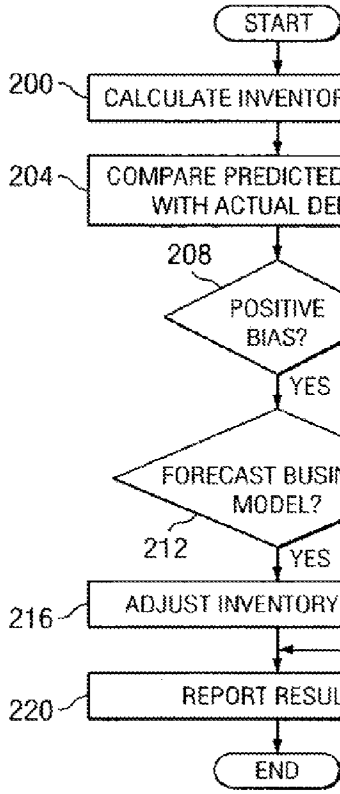


FIG. 6



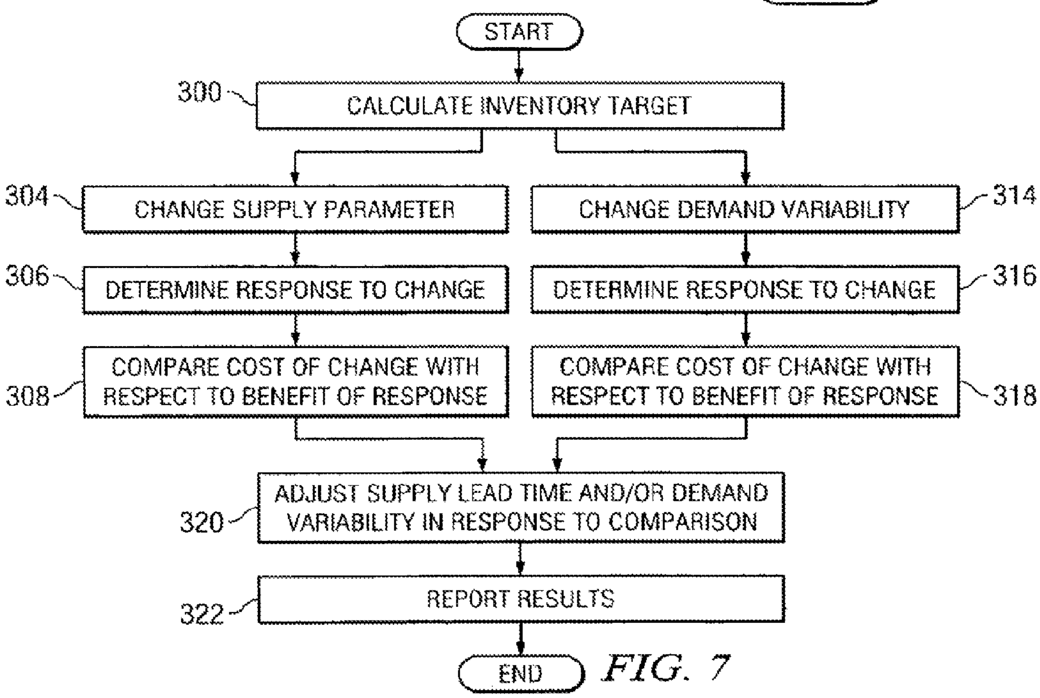


FIG. 7

DETERMINING AN INVENTORY TARGET FOR A NODE OF A SUPPLY CHAIN

CLAIM OF PRIORITY

This application is a continuation of Ser. No. 13/163,687, filed on Jun. 18, 2011 and entitled "Determining an Inventory Target for a Node of a Supply Chain," now U.S. Pat. No. 8,452,627 which is a continuation of U.S. Pat. No. 7,966,211, filed on Apr. 29, 2004 and entitled "Determining an Inventory Target for a Node of a Supply Chain," which claims priority under 35 U.S.C. §119(e) to U.S. Provisional No. 60/470,068, filed on May 12, 2003 and entitled "Strategic Inventory Optimization." U.S. Pat. Nos. 8,452,627 and 7,966,211 and U.S. Provisional No. 60/470,068 are commonly assigned to the assignee of the present application. The disclosure of related U.S. Pat. Nos. 8,452,627 and 7,966,211 and U.S. Provisional No. 60/470,068 are hereby incorporated by reference into the present disclosure as if fully set forth herein.

BACKGROUND

1. Technical Field of the Invention

This invention relates generally to the field of supply chain analysis and more specifically to determining an inventory target for a node of a supply chain.

2. Background of the Invention

A supply chain supplies a product to a customer, and may include nodes that store inventory such as parts needed to produce the product. A known technique for determining the proper amount of inventory at each node may involve predicting the amount of inventory needed at the nodes to satisfy customer demand. Known techniques for determining the proper amount of inventory, however, may not be able to accurately predict the amount of inventory needed at the nodes. It is generally desirable to accurately predict the amount of inventory needed at the nodes.

SUMMARY OF THE INVENTION

In accordance with the present invention, disadvantages and problems associated with previous supply chain analysis techniques may be reduced or eliminated.

According to one embodiment of the present invention, determining an inventory target for a node of a supply chain includes calculating a demand stock for satisfying a demand over supply lead time at the node of the supply chain, and calculating a demand variability stock for satisfying a demand variability of the demand over supply lead time at the node. A demand bias of the demand at the node is established. An inventory target for the node is determined based on the demand stock and the demand variability stock in accordance with the demand bias.

Certain embodiments of the invention may provide one or more technical advantages. For example, an inventory target may be determined from a demand stock and a demand variability stock. The demand stock covers mean demand over lead time, and the demand variability stock covers demand variability over lead time. Using the demand stock and the demand variability stock to determine an inventory target may provide for a more accuracy. Historical data may be used to determine the inventory target. The demand stock and the demand variability stock may be used to adjust parameters such as the supply lead time, the demand variability, or both to optimize the inventory target.

Certain embodiments of the invention may include none, some, or all of the above technical advantages. One or more other technical advantages may be readily apparent to one skilled in the art from the figures, descriptions, and claims included herein.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and its features and advantages, reference is made to the following description, taken in conjunction with the accompanying drawings, in which:

FIG. **1** is a block diagram illustrating an example system for determining an inventory target for a node of a supply chain;

FIG. **2** is a diagram illustrating an example supply chain that receives supplies from one or more suppliers and provides products to one or more customers;

FIG. **3** is a diagram illustrating an example node of the supply chain of FIG. **2**;

FIG. **4** is a graph illustrating a predicted demand and an actual demand with respect to time;

FIG. **5** is a flowchart illustrating an example method for determining an inventory target for a node of a supply chain;

FIG. **6** is a flowchart illustrating an example method for determining an inventory target for a node of a supply chain in accordance with historical data; and

FIG. **7** is a flowchart illustrating an example method for optimizing an inventory target for a node of a supply chain.

DETAILED DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram illustrating an example system **10** for determining an inventory target for a node of a supply chain that supplies products to customers in response to a demand. The demand may be represented as a predicted demand, which may be expressed as a mean demand and a demand variability. System **10** may, for example, calculate a demand stock and a demand variability stock for a node to satisfy the predicted demand. The demand stock covers mean demand over lead time, and the demand variability stock covers demand variability over lead time. System **10** may estimate an inventory target from the demand stock and the demand variability stock. According to one embodiment, system **10** may adjust the inventory target in response to historical data. According to another embodiment, system **10** may use the estimate of the inventory target to adjust parameters for the node such as the supply lead time, the demand variability, or both.

According to the illustrated embodiment, system **10** includes a client system **20**, a server system **24**, and a database **26** coupled as shown in FIG. **1**. Client system **20** allows a user to communicate with server system **24** to optimize inventory in a supply chain. Server system **24** manages applications for optimizing inventory in a supply chain. Database **26** stores data that may be used by server system **24**. According to the illustrated embodiment, server system **24** includes a processor **30** and one or more engines **32** coupled as shown in FIG. **1**. Processor **30** manages the operation of server system **24**, and may comprise any device operable to accept input, process the input according to predefined rules, and produce an output. According to the illustrated embodiment, engines **32** includes a demand planning engine **36**, an optimization engine **38**, and a supply chain planning engine **40**.

Demand planning engine **36** generates a demand forecast that predicts the demand at the nodes of a supply chain. Optimization engine **38** optimizes the inventory at the nodes of a supply chain, and may estimate an inventory target from a demand stock and a demand variability stock. Supply chain

planning engine **40** generates a plan for a supply chain. According to one embodiment, demand planning engine **36**, optimization engine **38**, and supply chain planning engine **40** may interact with each other. As an example, demand planning engine **36** may provide a demand forecast to optimization engine **38**. Optimization engine **38** may optimize the inventory in accordance with the demand forecast in order to generate inventory targets, which are provided to supply chain planning engine **40**. Supply chain planning engine **40** may generate a supply plan for the supply chain in accordance with the inventory targets.

According to one embodiment, optimization engine **38** may provide demand planning engine **36** and supply chain engine **40** with policy information. As an example, optimization engine may instruct demand planning engine **36** to decrease the demand variability of the demand estimate. As another example, optimization engine **38** may instruct supply chain planning engine **40** to decrease the supply lead time or supply lead time variation.

Client system **20** and server system **24** may each operate on one or more computers at one or more locations and may include appropriate input devices, output devices, mass storage media, processors, memory, or other components for receiving, processing, storing, and communicating information according to the operation of system **10**. For example, the present invention contemplates the functions of both client system **20** and server system **24** being provided using a single computer system, such as a single personal computer. As used in this document, the term "computer" refers to any suitable device operable to accept input, process the input according to predefined rules, and produce output, for example, a server, workstation, personal computer, network computer, wireless telephone, personal digital assistant, one or more microprocessors within these or other devices, or any other suitable processing device. Database **26** may include any suitable data storage arrangement and may operate on one or more computers at one or more locations.

Client system **20**, server system **24**, and database **26** may be integrated or separated according to particular needs. Client system **20**, server system **24**, and database **26** may be coupled to each other using one or more computer buses, local area networks (LANs), metropolitan area networks (MANs), wide area networks (WANs), a global computer network such as the Internet, or any other appropriate wireline, optical, wireless, or other links.

Modifications, additions, or omissions may be made to system **10** without departing from the scope of the invention. For example, system **10** may have more, fewer, or other modules. Moreover, the operations of system **10** may be performed by more, fewer, or other modules. For example, the operations of simulation engine **34** and optimization engine **38** may be performed by one module, or the operations of optimization engine **38** may be performed by more than one module. Additionally, functions may be performed using any suitable logic comprising software, hardware, other logic, or any suitable combination of the preceding. As used in this document, "each" refers to at least one member of a set.

FIG. **2** is a diagram illustrating an example supply chain **70** that receives supplies from one or more suppliers **80** and provides products to one or more customers **84**. Items flow through supply chain **70**, and may be transformed or remain the same as they flow through supply chain **70**. Items may comprise, for example, parts or supplies that may be used to generate the products. For example, an item may comprise a part of the product, or an item may comprise a supply that is used to manufacture the product, but does not become a part of the product. Downstream refers to the direction from suppliers **80** to customers **84**, and upstream refers to the direction from customers **84** to suppliers **80**.

Supply chain **70** may include any suitable number of nodes **76** and any suitable number of arcs **78** between nodes **76**, configured in any suitable manner. According to the illustrated embodiment, items from supplier **80** flow to node **76***a*, which sends items to node **76***b*. Node **76***b* sends items to node **76***c*, which sends items to nodes **76***d* and **76***e*. Nodes **76***d* and **76***e* provide products to customers **84***a* and **84***b*, respectively. A supply lead time for a node **76** refers to the time it takes for a supply to be provided to the node **76** from an upstream node **76**.

Although supply chain **70** is illustrated as having five nodes **76***a-e* and four arcs **78***a-d*, modifications, additions, or omissions may be made to supply chain **70** without departing from the scope of the invention. For example, supply chain **70** may have more or fewer nodes **76** or arcs **78**. Moreover, nodes **76** or arcs **78** may have any suitable configuration. For example, node **76***a* may supply items to node **76***c*, but not to node **76***b*.

Certain characteristics of supply chain **70** may make it difficult for supply chain **70** to respond to a customer demand. For example, high demand variability and long supply lead times may hinder the responsiveness of supply chain **70**. Redistributing inventory towards downstream nodes **76** of supply chain **70** may improve responsiveness. Distributing inventory towards downstream nodes **76**, however, may increase the inventory cost and the risk of obsolete inventory. Accordingly, different nodes **76** of supply chain **70** may be selected as response buffers in order to balance the responsiveness and flexibility of supply chain **70**.

FIG. **3** is a diagram illustrating an example node **76** of supply chain **70** of FIG. **2**. A demand forecast may be generated for node **76**. The demand forecast may predict a mean demand over supply lead time and a demand variability. A demand stock and a demand variability stock may be estimated for the mean demand over supply lead time and the demand variability. The demand stock of a node **76** represents the stock calculated to cover the mean demand over the supply lead time at the node **76**. The demand variability stock for a node **76** represents the stock calculated to cover the demand variability of the demand over the supply lead time at the node **76**. Since the mean demand over supply lead time is deterministic and the demand variability is probabilistic, the demand stock is deterministic and the variability stock is probabilistic.

For example, the supply lead time SLT for node **76** may be SLT=2 weeks, and the demand forecast may predict a mean demand d=1,000 units per week with a demand variability σ_d=10%. Optimization engine **38** may calculate demand stock SD=d×SLT=(1,000 units/1 week)×2 weeks=2,000 units. The demand variability stock SV may be calculated according to SD×σ_d=2,000 units×10%=200 units. The inventory target IT may be estimated from SD and SV according to IT=SD+SV=2,000 units+200 units=2,200 units.

According to one embodiment, optimization engine **38** may calculate the demand stock independently from the demand variability stock. Separate calculations of the demand stock and the demand variability stock may aid in identifying changes to a supply chain **70** that may be made. For example, if the demand stock is 85% of the target inventory, and the demand variability stock is 15% of the total inventory, then a user may determine that decreasing the demand stock may be more beneficial than decreasing the demand variability stock.

Separate calculations of the demand stock and the demand variability stock may also provide visibility on how changing certain parameters such as the supply lead time, supply lead

time variability, or demand variability affects optimization of inventory targets. For example, decreasing demand variability typically decreases the demand variability stock, which may allow for decreasing the inventory target, relaxing supply lead time requirements, or both. As another example, decreasing the supply lead time, supply lead time variability, or both typically decreases the demand stock, which may allow for decreasing the inventory target, relaxing demand variability requirements, or both.

FIG. **4** is a graph **90** illustrating a predicted demand **92** and an actual demand **94** with respect to time. Predicted demand **92** represents a demand that is calculated without knowledge of the actual demand, and may be determined from a demand forecast generated by demand planning engine **36**. Predicted demand **92** may include a mean demand d and a demand variability σ_d with respect to time. Actual demand **94** represents the known demand. In the illustrated example, actual demand **94** is greater than predicted demand **92**.

Different business models may use different types of demand forecasts or may not even use demand forecasts at all. Examples of business models include the build-to-forecast model, the assemble-to-order model, and the build-to-order model. According to the build-to-forecast model, products are produced in response to a demand forecast. Build-to-forecast models typically require an accurate and precise demand forecast. According to the assemble-to-order model, parts of the product may be produced, and then the product is assembled from the parts in response to an order. Assemble-to-order models typically require an accurate and precise demand forecast for the parts of the product. According to the build-to-order model, products are produced in response to an order from a customer rather than to a demand forecast.

FIG. **5** is a flowchart illustrating an example method for estimating an inventory target for a node **76** of supply chain **70**. The method begins at step **100**, where optimization engine **38** receives a demand forecast and a supply lead time for node **76**. Demand planning engine **36** may provide the demand forecast, and supply chain planning engine **40** may provide the supply lead time. The demand forecast may include a mean demand and a demand variability. According to one example, the supply lead time SLT may be SLT=2 weeks. The mean demand and the demand variability are established from the demand forecast at step **104**. According to one example, the demand forecast may predict a mean demand d=1,000 units per week with a demand variability σ_d=10%.

The demand stock is determined at step **108**. The demand stock may represent the stock that covers the mean demand over a supply lead time. The demand stock SD may be calculated by multiplying the mean demand d per time unit by the supply lead time SLT. For example, SD=d×SLT=(1,000 units/1 week)×2 weeks=2,000 units. The demand variability stock is determined at step **112**. The demand variability stock SV may be calculated by multiplying demand stock SD by variability (σ_d according to SV=SD×σ_d=2,000 units×10%=200 units. The inventory target is calculated at step **116**. The inventory target may be calculated by adding the demand stock with the demand variability stock. For example, inventory target IT may be estimated from SD and SV according to IT=SD+SV=2,000 units+200 units=2,200 units. The reports are resulted at step **120**. After reporting the results, the method ends.

Modifications, additions, or omissions may be made to the method without departing from the scope of the invention. Additionally, steps may be performed in any suitable order without departing from the scope of the invention.

FIG. **6** is a flowchart illustrating an example method for estimating inventory targets for supply chain **70** in accordance with historical data. The method begins at step **200**, where optimization engine **38** calculates an inventory target for a node **76** of supply chain **70**. The inventory target may be calculated according to the method described with reference to FIG. **5**. For example, the inventory target IT may be determined from a demand stock SD and a demand variability stock SV according to IT=SD+SV. Predicted demand **92** is compared with the actual demand **94** at step **204**. An example of predicted demand **92** and actual demand **94** is described with reference to FIG. **4**.

Predicted demand **92** may exhibit a demand bias such as a positive bias when compared with actual demand **94**. A demand bias refers to the tendency of predicted demand **92** to be greater than or less than actual demand **94**. A positive bias occurs when predicted demand **92** is less than actual demand **94**, and negative bias occurs when predicted demand **92** is greater than actual demand **94**. If there is no positive bias, the method proceeds to step to **220**.

If there is a positive bias, the method proceeds to step **212**. Supply chain **70** may be associated with a forecast business model such as a build-to-forecast or an assemble-to-order business model. If the business model is not a forecast business model, the method proceeds to step **220**. If the business model is a forecast business model, the method proceeds to step **216**. At step **216**, the inventory target is adjusted. The inventory target IT may be adjusted by, for example, ignoring the demand variability stock SV such that IT=SD. For a build-to-forecast business model, if the forecast for a product is positive, then the demand variability stock for the product might not be needed. For an assemble-to-order forecast, if the forecast for a part is positive, then the demand variability stock for the part might not be needed. The results are reported at step **220**. After reporting the results, the method ends.

Modifications, additions, or omissions may be made to the method without departing from the scope of the invention. Additionally, steps may be performed in any suitable order without departing from the scope of the invention.

FIG. **7** is a flowchart illustrating an example method for optimizing inventory in supply chain **70**. The method may be used to determine the effect of changing a supply parameter, a demand parameter, or both on inventory optimization. A supply parameter refers to a parameter relevant to the supply for a node **76** such as the supply lead time or the supply lead time variability. A demand parameter refers to a parameter relevant to the demand on a node **76** such as the mean demand or the demand variability. The benefits of the response may be compared with the cost of the change in order to adjust an inventory target. The method begins at step **300**, where optimization engine **38** calculates an inventory target. The inventory target may be calculated according to the method described with reference to FIG. **4**. For example, the inventory target may be calculated by adding a demand stock to a demand variability stock.

Steps **304** through **308** describe changing a supply parameter such as a supply lead time or a supply lead time variability and evaluating the effects of the change. The supply parameter is changed at step **304**. The supply parameter may be changed by, for example, decreasing the supply lead time. Changing a supply parameter however, typically has an associated cost. For example, costs related to decreasing a supply lead time may include an increase in delivery costs. The response to the change is determined at step **306**. The response may have an associated benefit. For example, decreasing the supply lead time may result in a decrease in the demand stock, which in turn results in a decrease in the

inventory target. The cost of the change is compared to the benefit of the response at step **308**. After comparing, the method proceeds to step **320**.

Step **314** through **318** describe changing the demand variability and evaluating the effects of the change. The demand variability is changed at step **314**. For example, the demand variability may be decreased by improving the precision of the demand forecast received from demand planning engine **36**. Changing the demand variability, however, may involve certain costs. For example, costs related to decreasing the demand variability may include the cost of purchasing software that generates a more precise demand estimate or the cost of increased time or data needed to produce a more precise demand estimate. The response to the change is determined at step **316**. The response may have an associated benefit. For example, decreasing the demand variability may decrease the demand variability stock, which in turn may decrease the inventory target. The cost of the change is compared to the benefit of the response to the change at step **318**. After comparing, the method proceeds to step **320**.

The supply lead time, the demand variability, or both are adjusted in response to the comparisons at step **320**. For example, if the benefit of changing the supply lead time outweighs the cost of changing the supply lead time, the supply lead time may be changed. As another example, if the benefit of changing the demand variability outweigh the cost of changing the demand variability, the demand variability may be changed. The results are reported at step **322**. After reporting the results, the method ends.

Modifications, additions, or omissions may be made to the method without departing from the scope of the invention. For example, steps **304** through **308** or steps **314** through **318** may be omitted. Additionally, steps may be performed in any suitable order without departing from the scope of the invention. For example, steps **304** through **308** and steps **314** through **318** may be preformed concurrently such that changing the supply lead time and the demand variability at steps **304** and **314** may be preformed concurrently. The responses may be checked substantially simultaneously at step **306** and **316**, and the costs and benefits may be compared substantially simultaneously at step **308** and **318**.

Certain embodiments of the invention may provide one or more technical advantages. For example, an inventory target may be determined from a demand stock and a demand variability stock. The demand stock covers mean demand over lead time, and the demand variability stock covers demand variability over lead time. Using the demand stock and the demand variability stock to determine an inventory target may provide for a more accuracy. Historical data may be used to determine the inventory target. The demand stock and the demand variability stock may be used to adjust parameters such as the supply lead time, the demand variability, or both to optimize the inventory target.

Although an embodiment of the invention and its advantages are described in detail, a person skilled in the art could make various alterations, additions, and omissions without departing from the spirit and scope of the present invention as defined by the appended claims.

What is claimed is:

1. A method of optimizing a supply chain parameter of a node of a supply chain, comprising:

calculating, by a computer, an inventory target for a node of a supply chain; by:

calculating a demand stock that satisfies a demand over supply lead time at a node of the supply chain;

calculating a demand variability stock that satisfies a demand variability of the demand over supply lead time at the node;

establishing the demand bias of the demand at the node; and

determining the inventory target of the node based on the demand stock and the demand variability stock in accordance with the demand bias;

changing, by the computer, a supply parameter associated with the node, the change in the supply parameter associated with a supply parameter cost or a supply parameter benefit;

adjusting, by the computer, the supply parameter associated with the node, when the benefit of the supply parameter exceeds the cost of the supply parameter; and

adjusting, by the computer, the inventory target of the node based on the adjusted supply parameter.

2. The method of claim **1**, further comprising:

changing, by the computer, a demand parameter associated with the node, the change in the demand parameter associated with a demand parameter cost or a demand parameter benefit;

adjusting, by the computer, the demand parameter associated with the node, when the benefit of the demand parameter exceeds the cost of the demand parameter; and

adjusting, by the computer, the inventory target of the node based on the adjusted demand parameter.

3. The method of claim **1**, wherein the supply parameter comprises the supply lead time or supply lead time variability.

4. The method of claim **2**, wherein the demand parameter comprises a mean demand or the demand variability.

5. The method of claim **1**, wherein calculating the inventory target of the node, comprises:

establishing that a predicted demand is less than an actual demand; and

using the demand stock but not the demand variability stock in determining the inventory target.

6. The method of claim **2**, further comprising:

changing a second demand parameter associated with the node, the change in the second demand parameter associated with a second demand cost and a second demand benefit;

changing a second supply parameter associated with the node, the change in the second supply parameter associated with a second supply cost and a second supply benefit;

determining the inventory target of the node according to the change in the supply parameter, the change in the demand parameter, and the comparison of the second demand cost and the second supply benefit.

7. The method of claim **2**, further comprising:

generating a demand forecast;

determining the demand over supply lead time from the demand forecast; and

determining the demand variability of the demand over supply lead time from the demand forecast.

8. The method of claim **2**, further comprising:

identifying a business model associated with the supply chain; and

adjusting the supply parameter and the demand parameter in accordance with the business model.

9. A system of optimizing a supply chain parameter of a supply chain, comprising:

an optimization engine tangibly embodied on a non-transitory computer-readable medium configured to calculate an inventory target for a node of a supply chain, comprising:

calculate a demand stock that satisfies a demand over supply lead time at a node of the supply chain;

calculate a demand variability stock that satisfies a demand variability of the demand over supply lead time at the node;

establish the demand bias of the demand at the node; and

determine the inventory target of the node based on the demand stock and the demand variability stock in accordance with the demand bias;

change a supply parameter associated with the node, the change in the supply parameter associated with a supply parameter cost or a supply parameter benefit;

adjust the supply parameter associated with the node, when the benefit of the supply parameter exceeds the cost of the supply parameter; and

adjust the inventory target of the node based on the adjusted supply parameter.

10. The system of claim **9**, further comprising:

change a demand parameter associated with the node, the change in the demand parameter associated with a demand parameter cost or a demand parameter benefit;

adjust the demand parameter associated with the node, when the benefit of the demand parameter exceeds the cost of the demand parameter; and

adjust the inventory target of the node based on the adjusted demand parameter.

11. The system of claim **9**, wherein the optimization engine is further configured to calculate the inventory target of the node, by:

establishing that a predicted demand is less than an actual demand; and

using the demand stock but not the demand variability stock in determining the inventory target.

12. The system of claim **10**, wherein the computer system is further configured to:

change a second demand parameter associated with the node, the change in the second demand parameter associated with a second demand cost and a second demand benefit;

change a second supply parameter associated with the node, the change in the second supply parameter associated with a second supply cost and a second supply benefit;

determine the inventory target of the node according to the change in the supply parameter, the change in the demand parameter, and the comparison of the second demand cost and the second supply benefit.

13. The system of claim **10**, wherein the computer system is further configured to:

generate a demand forecast;

determine the demand over supply lead time from the demand forecast; and

determine the demand variability of the demand over supply lead time from the demand forecast.

14. The system of claim **10**, wherein the computer system is further configured to:

identify a business model associated with the supply chain; and

adjust the supply parameter and the demand parameter in accordance with the business model.

15. A non-transitory computer-readable medium embodied with software for optimizing a supply chain parameter of a node of a supply chain, the software when executed by a computer is configured to:

calculate an inventory target for a node of a supply chain; comprising:

calculate a demand stock that satisfies a demand over supply lead time at a node of the supply chain;

calculate a demand variability stock that satisfies a demand variability of the demand over supply lead time at the node;

establish the demand bias of the demand at the node; and

determine the inventory target of the node based on the demand stock and the demand variability stock in accordance with the demand bias;

change a supply parameter associated with the node, the change in the supply parameter associated with a supply parameter cost or a supply parameter benefit;

adjust the supply parameter associated with the node, when the benefit of the supply parameter exceeds the cost of the supply parameter; and

adjust the inventory target of the node based on the adjusted supply parameter.

16. The non-transitory computer-readable medium software of claim **15**, wherein the software is further configured to:

change a demand parameter associated with the node, the change in the demand parameter associated with a demand parameter cost or a demand parameter benefit;

adjust the demand parameter associated with the node, when the benefit of the demand parameter exceeds the cost of the demand parameter; and

adjust the inventory target of the node based on the adjusted demand parameter.

17. The non-transitory computer-readable medium software of claim **15**, wherein the software calculates the inventory target of the node, by:

establishing that a predicted demand is less than an actual demand; and

using the demand stock but not the demand variability stock in determining the inventory target.

18. The non-transitory computer-readable medium software of claim **16**, wherein the software is further configured to:

change a second demand parameter associated with the node, the change in the second demand parameter associated with a second demand cost and a second demand benefit;

change a second supply parameter associated with the node, the change in the second supply parameter associated with a second supply cost and a second supply benefit;

determine the inventory target of the node according to the change in the supply parameter, the change in the demand parameter, and the comparison of the second demand cost and the second supply benefit.

19. The non-transitory computer-readable medium software of claim **16**, wherein the software is further configured to:

generate a demand forecast;

determine the demand over supply lead time from the demand forecast; and

determine the demand variability of the demand over supply lead time from the demand forecast.

20. The non-transitory computer-readable medium software of claim **16**, wherein the software is further configured to:

identify a business model associated with the supply chain; and

adjust the supply parameter and the demand parameter in accordance with the business model.

* * * * *

EXHIBIT 6

US010572856B2

(12) United States Patent
Rau et al.

(10) Patent No.: US 10,572,856 B2
(45) Date of Patent: Feb. 25, 2020

(54) **CUSTOM APPLICATION BUILDER FOR SUPPLY CHAIN MANAGEMENT**

(75) Inventors: **Anand Rau**, Irvine, CA (US); **Tarak Patel**, Corona, CA (US)

(73) Assignee: **JDA Software Group, Inc.**, Scottsdale, AZ (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 680 days.

(21) Appl. No.: **11/371,985**

(22) Filed: **Mar. 8, 2006**

(65) **Prior Publication Data**

US 2006/0206411 A1 Sep. 14, 2006

Related U.S. Application Data

(60) Provisional application No. 60/660,543, filed on Mar. 9, 2005.

(51) **Int. Cl.**
G06Q 10/00 (2012.01)
G06Q 10/10 (2012.01)
G06Q 10/08 (2012.01)

(52) **U.S. Cl.**
CPC ***G06Q 10/10*** (2013.01); ***G06Q 10/08*** (2013.01); *G06Q 10/0835* (2013.01)

(58) **Field of Classification Search**
CPC G06Q 10/083; G06Q 10/06; G06Q 10/08; G06Q 10/0835; G06Q 10/10
USPC 705/7
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,603,034	A *	2/1997	Swanson	717/111
5,754,858	A *	5/1998	Broman et al.	717/111
5,950,001	A *	9/1999	Hamilton et al.	717/107
6,308,178	B1 *	10/2001	Chang et al.	
6,957,191	B1 *	10/2005	Belcsak et al.	705/38
7,058,587	B1 *	6/2006	Horne	705/7.22
7,325,015	B2 *	1/2008	Zhou et al.	707/104.1
7,577,907	B2 *	8/2009	Vishnia-Shabtai et al.	715/255
2002/0111953	A1 *	8/2002	Snyder	707/101
2002/0188486	A1 *	12/2002	Gil et al.	705/7
2003/0074247	A1 *	4/2003	Dick et al.	705/9
2004/0254825	A1 *	12/2004	Hsu et al.	705/8
2005/0021348	A1 *	1/2005	Chan et al.	705/1
2005/0262453	A1 *	11/2005	Massasso	715/965
2006/0101391	A1 *	5/2006	Ulke et al.	717/107
2006/0106897	A1 *	5/2006	Sapozhnikov et al.	707/204
2006/0190486	A1 *	8/2006	Zhou et al.	707/104.1

(Continued)

OTHER PUBLICATIONS

Rossi, G., Schwabe, D., & Guimaraes, R. (2001). Designing Personalized Web Applications. In Conference proceedings: the Tenth International World Wide Web Conference, Hong Kong, May 1-5, 2001 (pp. 275-284). New York: ACM.*

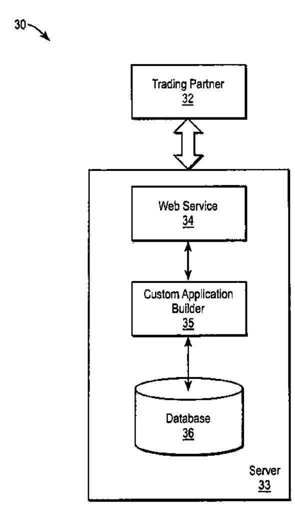
Primary Examiner — William S Brockington, III

(74) *Attorney, Agent, or Firm* — Spencer Fane LLP; Steven J. Laureanti

(57) **ABSTRACT**

Supply chain management is performed by building a user-defined custom application that uses supply chain data. The supply chain data is returned for display and/or use in the custom application based on metadata stored in a database that is arranged to store supply chain data. The metadata specifies a stored procedure that when called based on parameters in a request generated in response to a user selecting a custom application, returns the supply chain data that may be used in whole or in part in the custom application.

23 Claims, 15 Drawing Sheets



(56) **References Cited**

U.S. PATENT DOCUMENTS

2006/0190714 A1* 8/2006 Vaszary G06F 9/5083
713/100

* cited by examiner

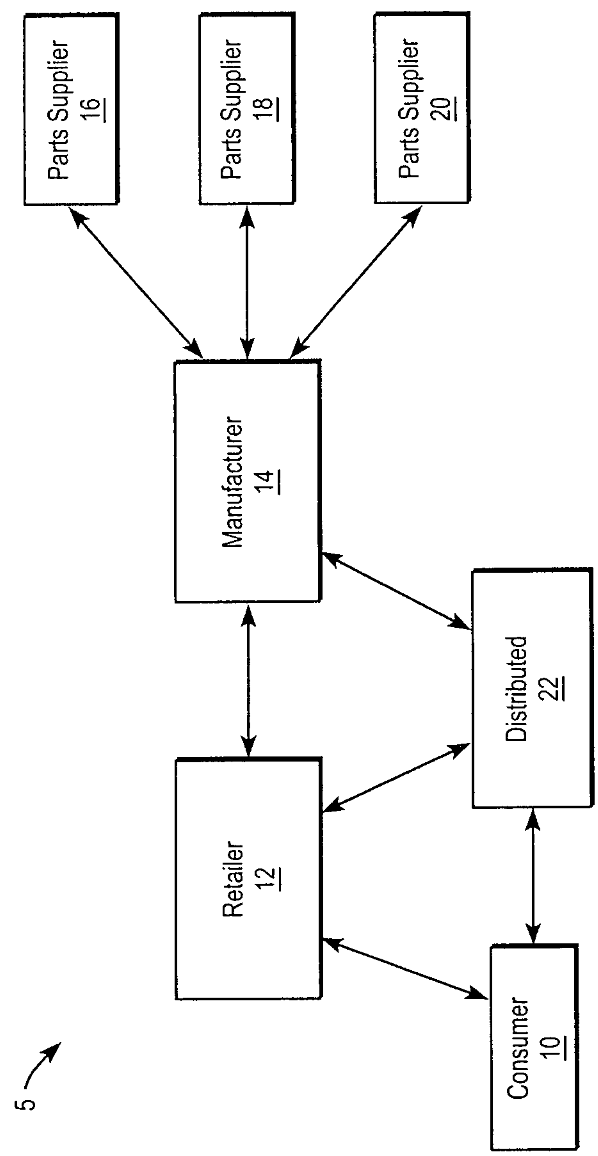


FIG. 1
(PRIOR ART)

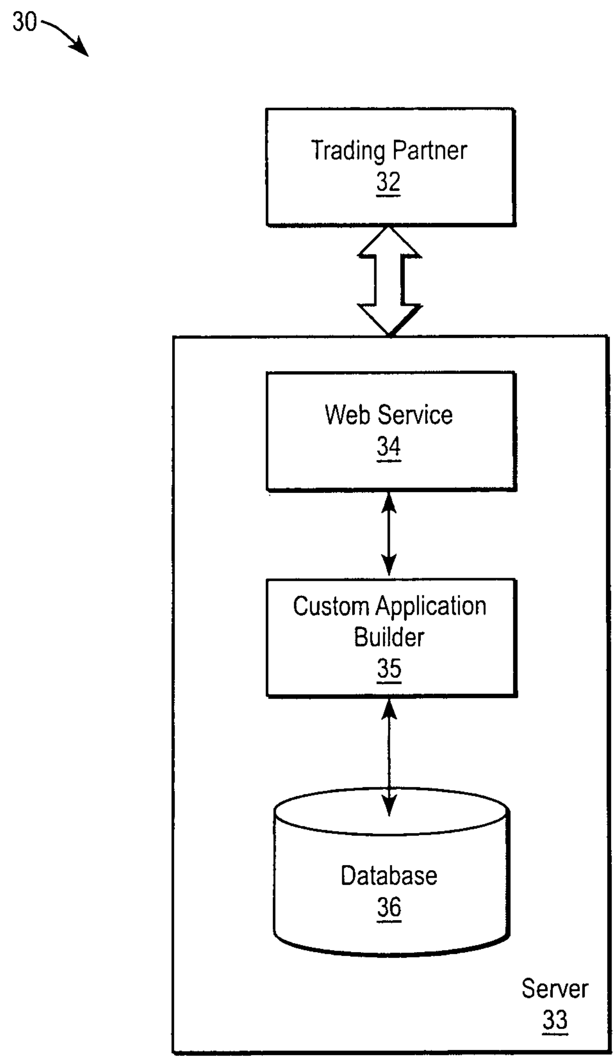

30
Trading Partner
32
Web Service
34
Custom Application Builder
35
Database
36
Server
33

FIG. 2

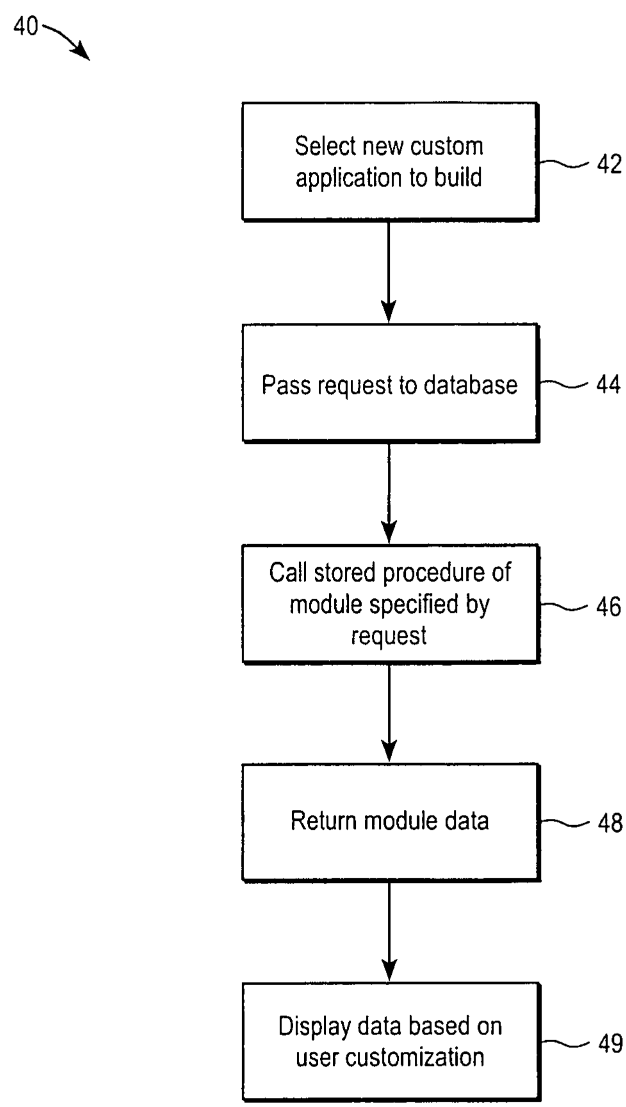

40
Select new custom application to build
42
Pass request to database
44
Call stored procedure of module specified by request
46
Return module data
48
Display data based on user customization
49

FIG. 3

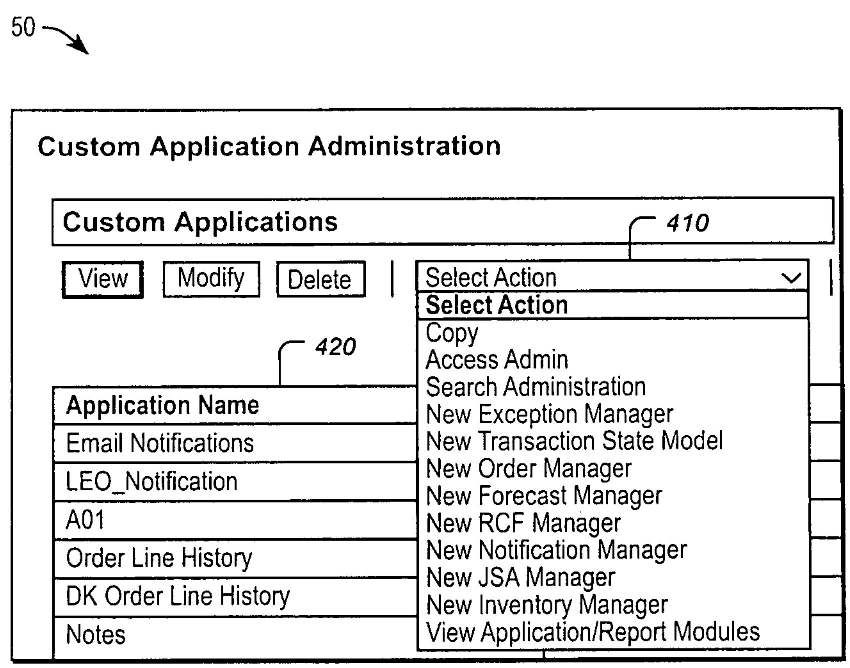

50
Custom Application Administration
Custom Applications
410
View
Modify
Delete
Select Action
Select Action
Copy
Access Admin
Search Administration
New Exception Manager
New Transaction State Model
New Order Manager
New Forecast Manager
New RCF Manager
New Notification Manager
New JSA Manager
New Inventory Manager
View Application/Report Modules
420
Application Name
Email Notifications
LEO_Notification
A01
Order Line History
DK Order Line History
Notes

FIG. 4

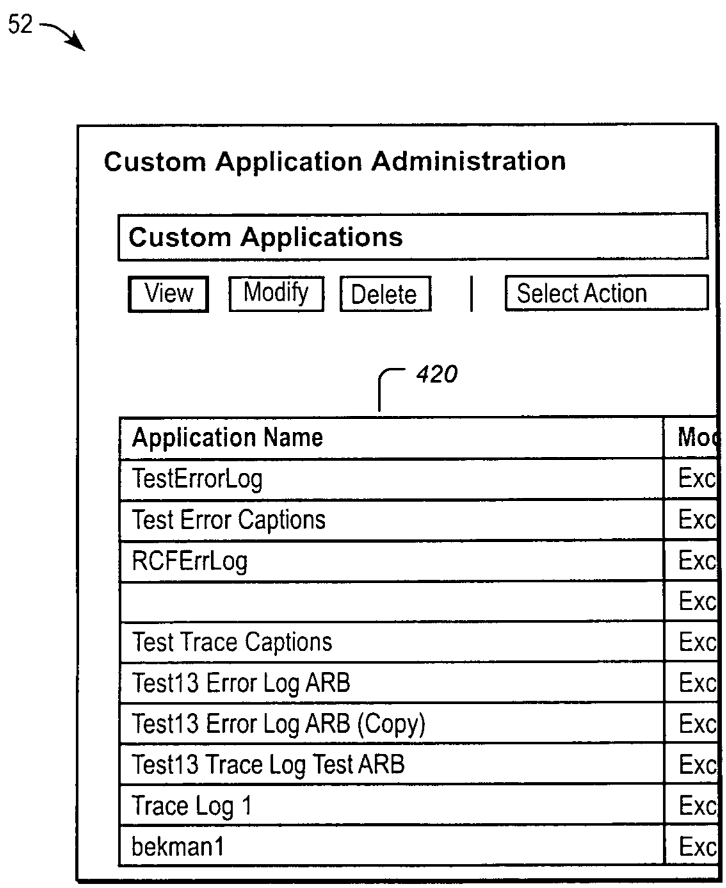

52
Custom Application Administration
Custom Applications
View
Modify
Delete
Select Action
420
Application Name
TestErrorLog
Test Error Captions
RCFErrLog
Test Trace Captions
Test13 Error Log ARB
Test13 Error Log ARB (Copy)
Test13 Trace Log Test ARB
Trace Log 1
bekman1

FIG. 5

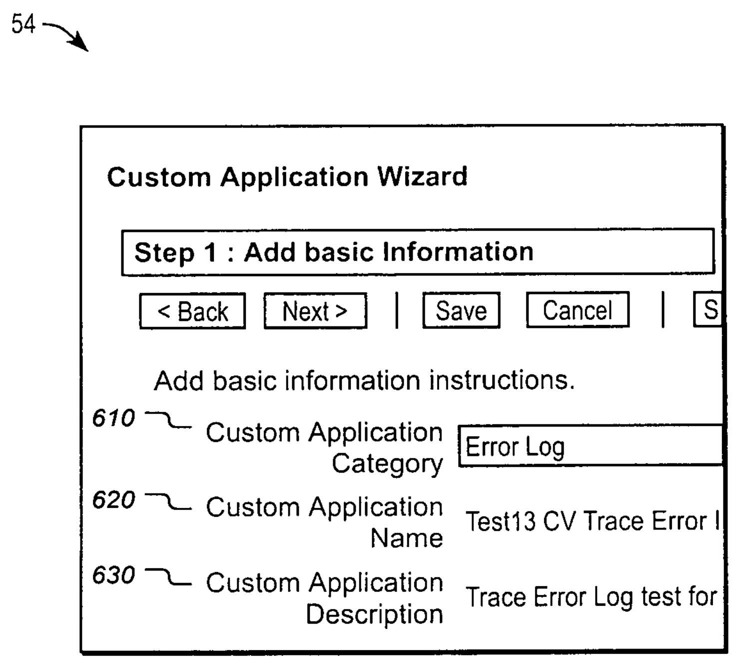

54
Custom Application Wizard
Step 1 : Add basic Information
< Back
Next >
Save
Cancel
S
Add basic information instructions.
610
Custom Application Category
Error Log
620
Custom Application Name
Test13 CV Trace Error l
630
Custom Application Description
Trace Error Log test for

FIG. 6

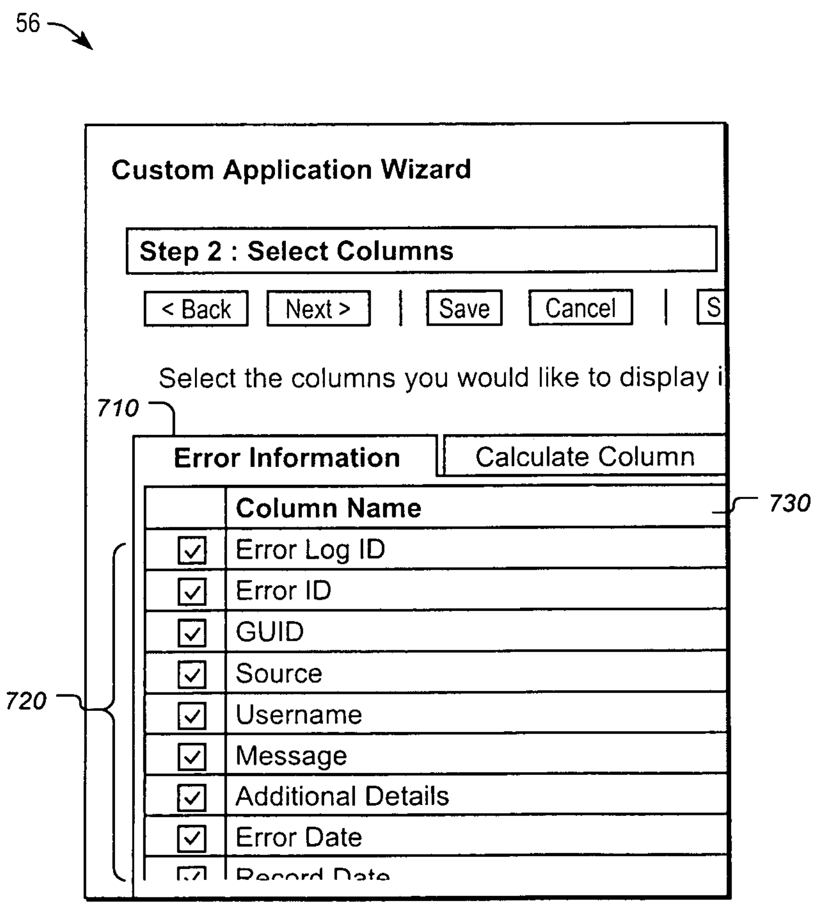

56
Custom Application Wizard
Step 2 : Select Columns
< Back
Next >
Save
Cancel
S
Select the columns you would like to display i
710
Error Information
Calculate Column
730
Column Name
Error Log ID
Error ID
GUID
Source
Username
Message
Additional Details
Error Date
Record Date
720

FIG. 7

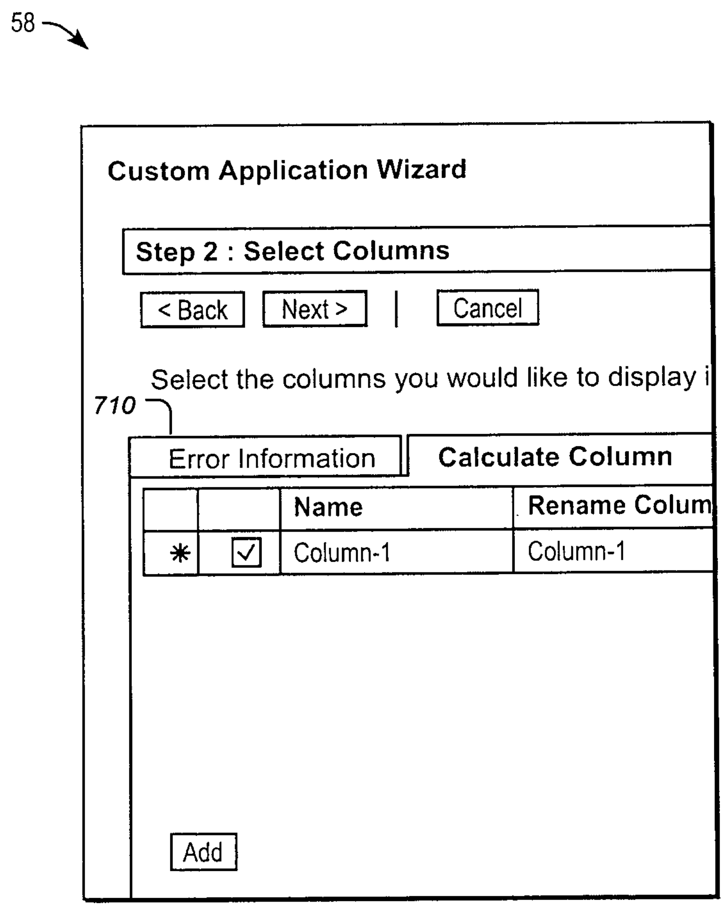

58
Custom Application Wizard
Step 2 : Select Columns
< Back
Next >
Cancel
Select the columns you would like to display i
710
Error Information
Calculate Column
Name
Rename Colum
Column-1
Column-1
Add

FIG. 8

60

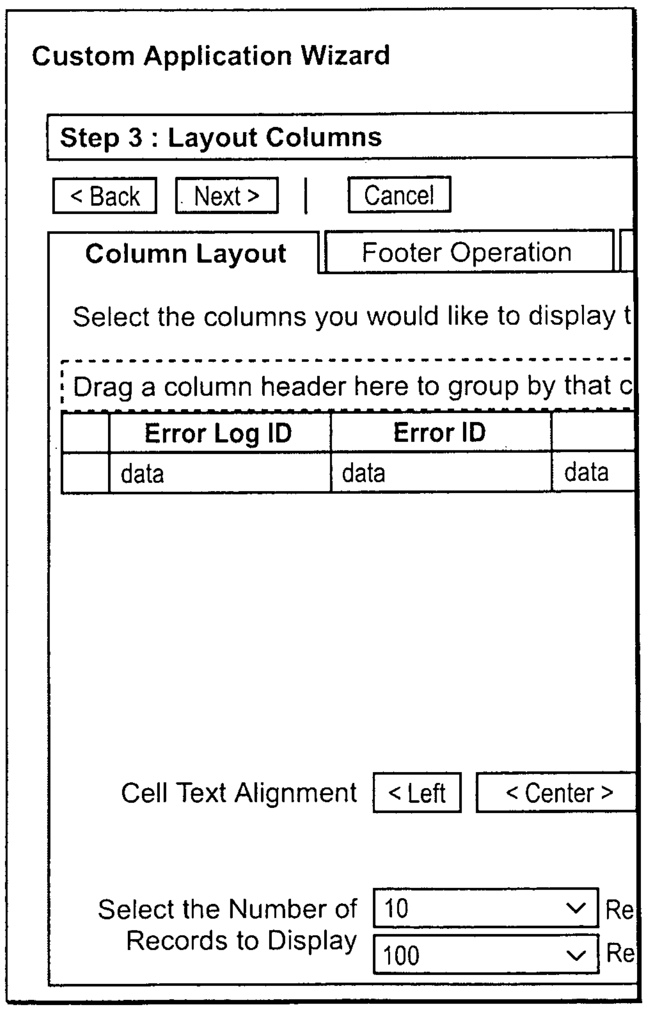

Custom Application Wizard
Step 3 : Layout Columns
< Back
Next >
Cancel
Column Layout
Footer Operation
Select the columns you would like to display t
Drag a column header here to group by that c
Error Log ID
Error ID
data
data
data
Cell Text Alignment
< Left
< Center >
Select the Number of Records to Display
10
Re
100
Re

FIG. 9

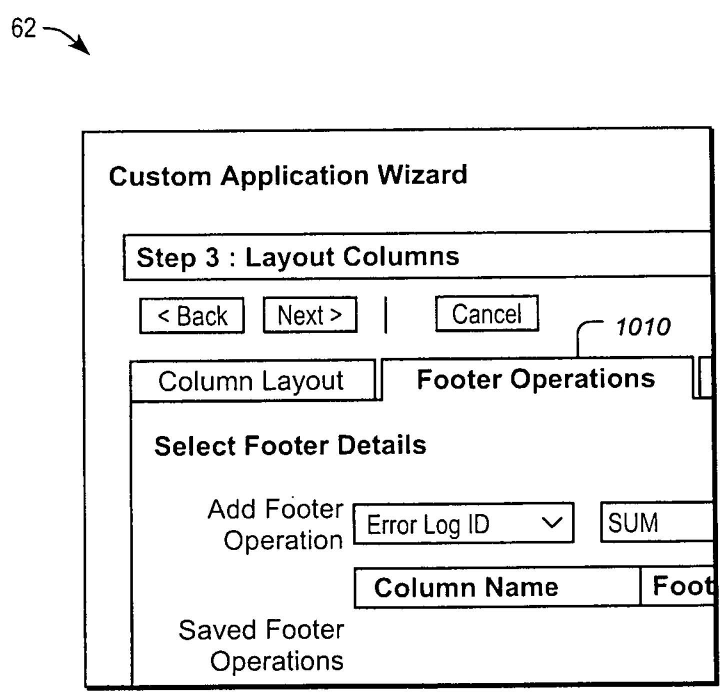

62
Custom Application Wizard
Step 3 : Layout Columns
< Back
Next >
Cancel
1010
Column Layout
Footer Operations
Select Footer Details
Add Footer Operation
Error Log ID
SUM
Column Name
Foot
Saved Footer Operations

FIG. 10

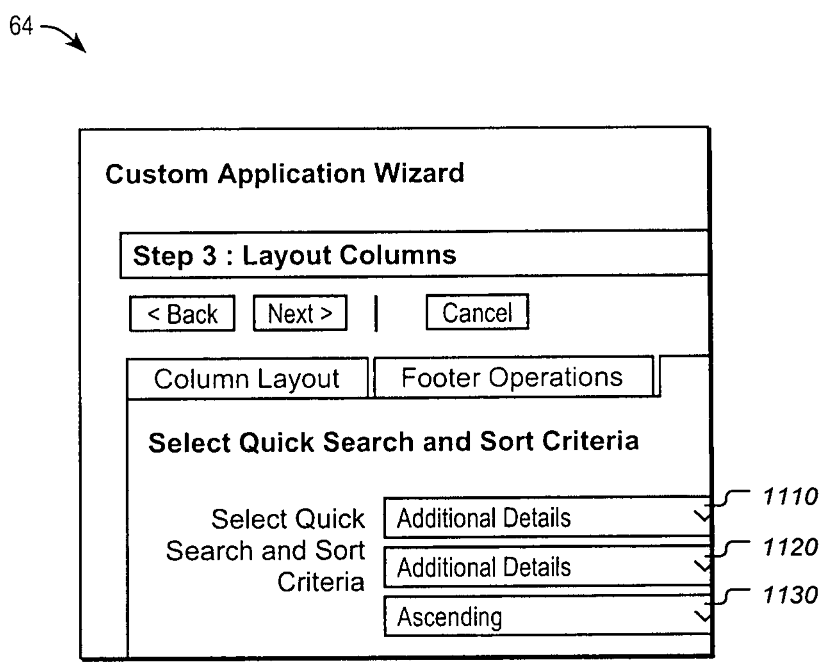

64
Custom Application Wizard
Step 3 : Layout Columns
< Back
Next >
Cancel
Column Layout
Footer Operations
Select Quick Search and Sort Criteria
Select Quick Search and Sort Criteria
Additional Details
Additional Details
Ascending
1110
1120
1130

FIG. 11

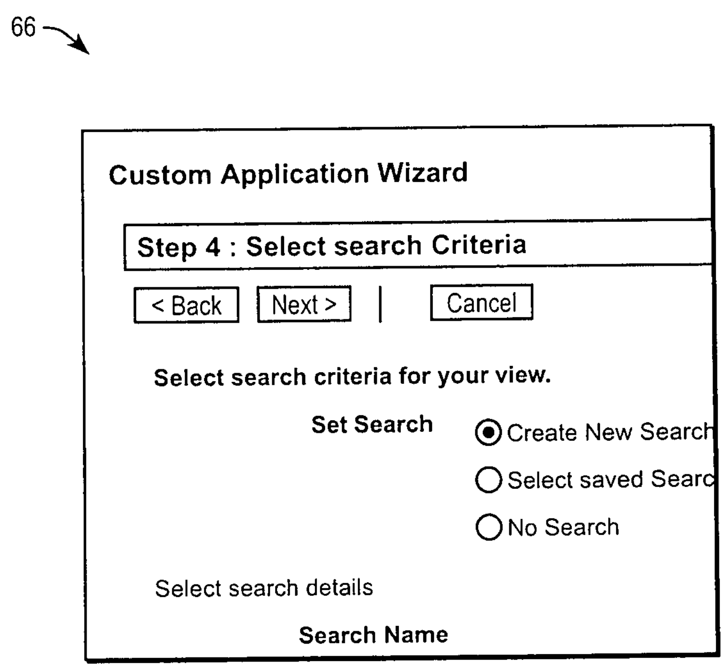

66
Custom Application Wizard
Step 4 : Select search Criteria
< Back
Next >
Cancel
Select search criteria for your view.
Set Search
Create New Search
Select saved Searc
No Search
Select search details
Search Name

FIG. 12

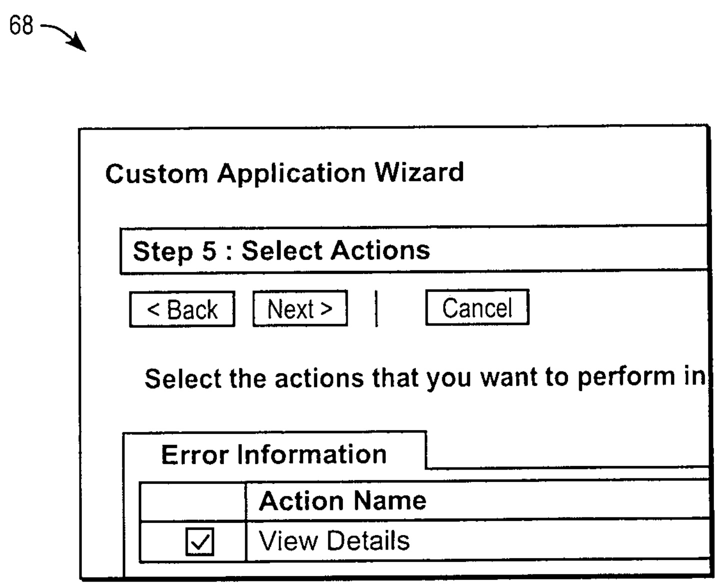

68
Custom Application Wizard
Step 5 : Select Actions
< Back
Next >
Cancel
Select the actions that you want to perform in
Error Information
Action Name
View Details

FIG. 13

70

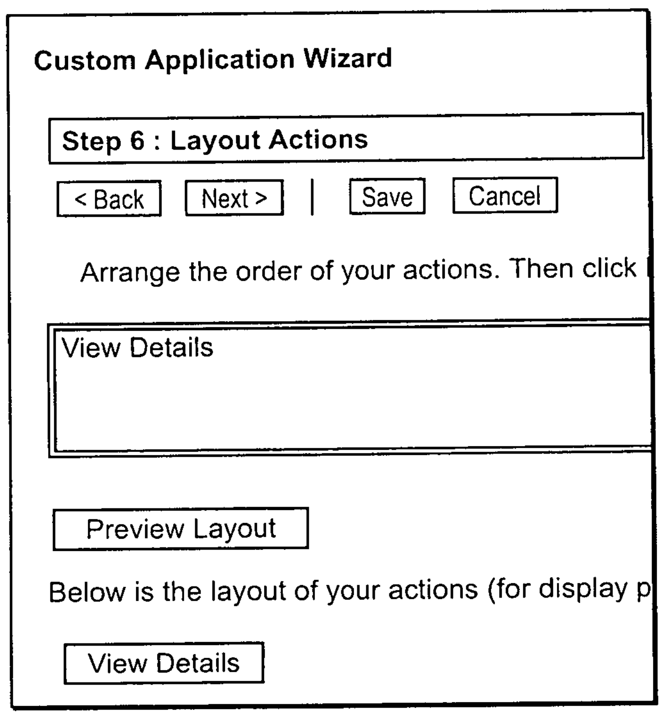

Custom Application Wizard
Step 6 : Layout Actions
< Back
Next >
Save
Cancel
Arrange the order of your actions. Then click
View Details
Preview Layout
Below is the layout of your actions (for display p
View Details

FIG. 14

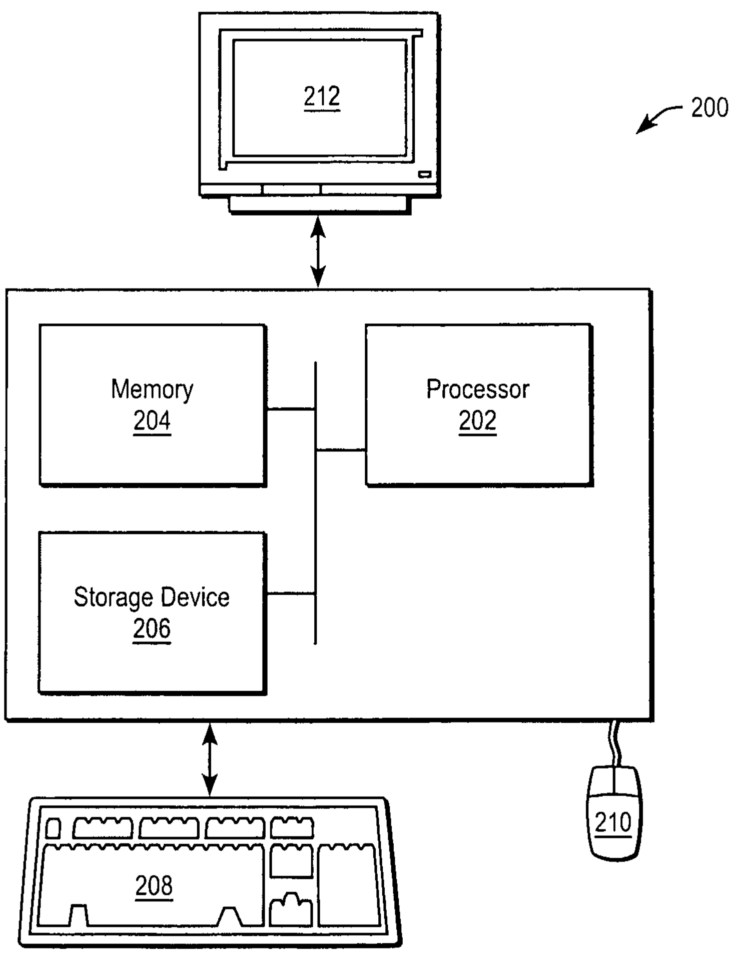


FIG. 15

CUSTOM APPLICATION BUILDER FOR SUPPLY CHAIN MANAGEMENT

CROSS REFERENCE TO RELATED APPLICATIONS

This application claims priority under 35 U.S.C. § 119(e) to U.S. Provisional Patent Application No. 60/660,543, filed on Mar. 9, 2005 and entitled "Custom Application Builder for Supply Chain Management", the entirety of which is hereby incorporated by reference. Further, the subject matter of the present application may be related to subject matter in the following commonly owned applications: U.S. patent application Ser. No. 11/158,371, filed on Jun. 22, 2005 and entitled "Program-Based Supply Chain Management", which claims the benefit of U.S. Provisional Patent Application No. 60/659,829, filed on Mar. 8, 2005 and entitled "Program-Based Supply Chain Management" and U.S. patent application Ser. No. 11/371,960, filed on Mar. 8, 2006 and entitled "Configuration State Model for Supply Chain Management", which claims the benefit of U.S. Provisional Patent Application No. 60/660,747, filed on Mar. 10, 2005 and entitled "Configurable State Model for Supply Chain Management".

FIELD OF ART

The present invention generally relates to supply chain management applications and more particularly relates to customizing the display and use of supply chain data in supply chain management applications.

BACKGROUND

In general, a "supply chain" relates to the flow of goods, services, and/or associated information from a source to a consumer. The supply chain may include various entities, any one or combination of which are, for example, capable of: receiving an order from a consumer; processing the order; managing payment for the order; determining the resources needed to fulfill the order; determining a course of action to fulfill the order; managing the flow of and/or distributing information regarding the order; managing resources needed to fulfill the order; manufacturing and/or integrating components of the order (e.g., assembling the parts of an ordered good); distributing the ordered good and/or service to the consumer; and forecasting needs and sales of future orders based on previous orders. Those skilled in the art will note that distributing responsibilities and functions over various entities in a supply chain may result in overall or targeted cost savings, improved order processing efficiency (e.g., improved delivery times), and/or improved quality of ordered goods and services.

FIG. **1** shows a typical supply chain **5**. Order flow in the supply chain **5** generally begins when a consumer **10** places an order with a retailer **12**. The retailer **12** may be a "brick-and-mortar" store or an "online" outfit accessible via the Internet. The retailer **12** receives the order from the consumer **10**, checks the order for errors, and ensures payment for the order. Payment for the order may occur through direct payment at the time the order is placed, payment based on a line of credit established for the consumer **10**, or charge for the order at the time the order is fulfilled and shipped to the consumer **10**.

The retailer **12**, based on the placed order, forwards the order to a manufacturer **14**. The manufacturer **14**, based on the needs of the placed order, either uses "on-hand" parts or obtains them from any one or more of parts suppliers (e.g., wholesalers) **16**, **18**, **20**. Upon manufacture of the ordered good, a distributor **22** delivers or causes the delivery of the ordered good to the consumer **10**.

Further, as discernible from FIG. **1**, information regarding the placed order and status thereof may be distributed in any variation across the entities shown in FIG. **1**. For example, the distributor **22** may inform the retailer **12** when the ordered good is shipped to the consumer **10**.

Managing the flow of goods, materials, services, and/or associated information in a supply chain, such as that shown in FIG. **1**, is referred to as "supply chain management" (SCM). In general, supply chain management involves managing the design, planning, execution, control, and monitoring of processes and activities of a supply chain. Such management is generally performed with the objectives of, for example, reducing inventory, increasing delivery times, reducing costs, increasing sales, synchronizing supply with current and forecasted demand, and/or increasing overall order processing efficiency.

SUMMARY

According to at least one aspect of one or more embodiments of the present invention, a computer-implemented method for building a custom application for supply chain management comprises: storing application parameters for a trading partner in a supply chain, where the application parameters describe at least one customized display of supply chain data for the trading partner based on a role of the trading partner; receiving a request for supply chain data from the trading partner; and responsive to the request, providing the requested supply chain data to the trading partner in accordance with the stored application parameters.

According to at least one other aspect of one or more embodiments of the present invention, a supply chain management system for building a custom application comprises: a web interface arranged to allow a trading partner to provide application parameters for the custom application; a custom application builder module arranged to generate a request for supply chain data based on the provided application parameters; and a database arranged to store the supply chain data, where the database is arranged to return the supply chain data to the custom application builder module based on parameters specified in the request.

According to at least one other aspect of one or more embodiments of the present invention, a method of supply chain management comprises: receiving a request from a user for supply chain data stored in a database, where the request comprises parameters specifying a category of the supply chain data needed for an application selected by the user; invoking a procedure to return the supply chain data from the database to the user at least partly based on the parameters in the request; and returning the supply chain data for display and use by the application according to customization preferences selected by the user.

According to at least one other aspect of one or more embodiments of the present invention, a method of supply chain management comprises: selecting a type of application for viewing supply chain data, where the supply chain data is stored in a database; generating a web-based request for the supply chain data based on the selecting; and selectively customizing a display of at least a portion of the supply chain data returned from the database in response to the web-based request.

According to at least one other aspect of one or more embodiments of the present invention, a supply chain management system comprises: a user system accessibly by a user, where the user system provides the user a capability to build a custom application requiring supply chain data; a web interface accessible by the user system, where the web interface is arranged to handle a request from the user system for the supply chain data; and a database arranged to store the supply chain data, where a procedure in metadata stored in the database is invocable to return the supply chain data based on parameters in the request.

According to at least one other aspect of one or more embodiments of the present invention, a computer-readable medium having instructions recorded therein to: render a web-based application for allowing a user to select an application to customize, where the application requires supply chain data; receive a request from the user to view supply chain data using the application; authenticate the request; transmit the authenticated request to a database having a procedure stored therein, the procedure invocable to return supply chain data specified by parameters in the request; and provide the user with at least one of selectable and definable preferences for the application to view and use at least a portion of the returned supply chain data.

The features and advantages described herein are not all inclusive, and, in particular, many additional features and advantages will be apparent to one skilled in the art in view of the drawings, specification, and claims. Moreover, it should be noted that the language used in the specification has been principally selected for readability and instructional purposes and may not have been selected to circumscribe the claimed invention.

BRIEF DESCRIPTION OF DRAWINGS

FIG. **1** shows a block diagram of a typical supply chain.

FIG. **2** shows a block diagram of a supply chain management system in accordance with an embodiment of the present invention.

FIG. **3** shows a flow process for supply chain management in accordance with an embodiment of the present invention.

FIG. **4** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **5** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **6** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **7** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **8** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **9** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **10** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **11** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **12** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **13** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **14** shows a screenshot of a user interface for building a custom application in accordance with an embodiment of the present invention.

FIG. **15** shows a computer system in accordance with an embodiment of the present invention.

DETAILED DESCRIPTION

In the following description of embodiments of the present invention, numerous specific details are set forth in order to provide a more thorough understanding of the present invention. However, it will be apparent to one skilled in the art that the present invention may be practiced without one or more of these specific details. In other instances, well-known features have not been described in detail to avoid unnecessarily complicating the description.

Generally, embodiments of the present invention relate to methods and systems for supply chain management. In one or more embodiments, supply chain management software allows a user to build custom applications for interacting with supply chain related data.

FIG. **2** shows at least a portion of a supply chain management system **30** in accordance with an embodiment of the present invention. A user at trading partner (e.g., consumer, retailer, manufacturer, parts supplier, distributor) **32** desiring to interact with (e.g., view, act upon) supply chain related data may build a “custom” application using data in one or more particular “modules” (“customization” further described below with reference to FIGS. **3-15**). A module in system **30** may relate to any aspect of supply chain management. For example, a module may relate to purchase order data, inventory data, forecasting data, invoice data, and/or management data.

Accessing and customizing the use and display of module data involves passing a request from the trading partner **32** to a supply chain management server **33** (via, for example, an HTML application (i.e., the trading partner **32** may communicate with the supply chain management server **33** over the Internet)). The request generally includes application parameters as to, for example, the type of application the trading partner **32** wishes to customize. Further, in one or more embodiments, the application parameters may be dependent on or relate to a role of trading partner **32** in the supply chain. For example, a distributor may request module data that specifically pertains to the function of the distributor.

In one or more embodiments, a request for module data by trading partner **32** may be generated by a software application resident at the trading partner **32**. In one or more other embodiments, the trading partner **32** may access a web-based application to generate the request, where the web-based application is rendered by a web service **34**. Regardless of how the request for module data is generated, the web service **34** may require identification information to authenticate the user at trading partner **32**. The provided identification information may further indicate what permissions the user at trading partner **32** has to access particular module data. For example, the user at trading partner **32** may have permission to access purchase order data but may not have permission to access data relating to administrative functions.

Further, those skilled in the art will note that web service **34** may support various platforms. For example, in one or more embodiments, web service **34** may support both Microsoft Windows®-based clients and Apple Macintosh® operating system-based clients.

Once authenticated, the module data request is passed from the web service **34** to custom application builder module **35**. The custom application builder module **35** generates one or more messages based on the module data request. In one or more embodiments, these messages may be coded in an extensible markup language (XML) format.

The messages generated by the custom application builder module **35** are passed to a database **36**. Although database **36** is shown in FIG. **2** as being part of the supply chain management server **33**, those skilled in the art will note that in one or more embodiments, database **36** may not be part of the supply chain management server **33**. Instead, database **36** may be operatively connected to the supply chain management server **33** via, for example, a network connection.

Database **36** contains definitions of various modules. The definition of a module (a "module definition file") specifies, for example, (i) the location of data in the database **36** for the module and (ii) a stored procedure to invoke when an application of the module is instantiated. Further, in one or more embodiments, the definition of a module may be described in metadata, e.g., an XML file. Thus, from a programming perspective, a developer specifies the appropriate metadata, stores the metadata in the database **36**, and creates a method that effectively allows a user-level application to extract data from the database **36** to the module used to implement the user-level application. Further, in one or more embodiments, a new definition of a module or a definition of a new module may be uploaded to the database **36** at any time.

The stored procedure of a module is called in response to receiving a message for data of the module from the custom application builder module **35**. When the stored procedure is called, data associated with the corresponding module is returned from the database **36** to the custom application builder module **35**. In general, the custom application builder module **35** generates and renders user interfaces (UIs) that allow a user at trading partner **32** to interact with supply chain data via web service **34**. As further described below with reference to FIGS. **3-15**, the display and/or use of module data returned to the custom application builder module **35** is customizable by the user at trading partner **32** via web service **34**.

Further, in one or more embodiments, supply chain data stored in the database **36** may be updated (e.g., removed, modified, added, reorganized) based on activities in a supply chain associated with the database **36**. Further still, in one or more embodiments, a user building a custom application via, for example, web service **34**, may additionally directly update or indirectly cause to update supply chain data stored in the database **36**.

FIG. **3** shows a flow process **40** for supply chain management in accordance with an embodiment of the present invention. Initially, a user, via, for example, a web interface, selects a new custom application to build **42**. Based on the selection, the web interface generates an XML message that is passed to a database **44**, which in turn calls a stored procedure of the module specified by parameters in the XML message **46**. Thereafter, the appropriate module data is returned by the database **48** and displayed to the user, via the web interface, based on the user's customization of the selected application **49**. The user's customization selections may be selected via the web interface so that when supply chain data is returned from the database, that supply chain data may be displayed according to the user's customization preferences/selections.

As described above with reference to FIGS. **2** and **3**, a user may build custom applications for interacting with module data. FIGS. **4-14** show user interfaces that may be used to build one or more custom supply chain management application. FIG. **4** shows a screenshot **50** of a user interface that allows the user to select a type of custom application to build. As shown in FIG. **4**, the user can select using a pull-down menu any one of various actions to create a new custom application.

Further, in one or more embodiments, the application types shown in FIG. **4** may be associated with "child" applications (not shown) that may be selected by the user. In such embodiments, the user may create a child application and then link the child application to its parent.

Customizing Applications

In general, in one or more embodiments, a user may customize the display and use of module data returned from a database (e.g., database **36** in FIG. **2**). The customized display may provide for any combination of the following: a selection of columns to display and one or more user-defined column names; user-defined column widths and alignments; a selection of default sort criteria and sort direction; a selection of default search criteria (the search criteria may allow for the searching of data in a displayed column); a user-defined custom search; user selected action buttons and the ability to rename and rearrange the actions; and the ability to assign another custom display to an action button.

Using the user interface shown in FIG. **4**, the user may select (from the dropdown menu **410** shown in FIG. **4**) a new type of custom application to create or the user may select (from the grid **420** shown in FIG. **4**) an existing custom application to modify. Once a custom application is selected as shown in FIG. **4**, a user may then begin to customize the selected application. For example, in the screenshot of user interface **52** shown in FIG. **5**, customization occurs by first accessing a user interface that displays a list of applications. From this list, the user selects one of the named applications to modify. Further, although not shown in FIG. **5**, additional information may be presented to the user when selecting an application to modify. For example, information may be provided relating to the name of the module that implements the application, the date and time of the last modification of the application, and a last user who modified the application.

Once an application is selected for customization, a user interface, such as that of the screenshot of user interface **54** shown in FIG. **6**, is displayed to allow the user to add, remove, or edit information about the selected application. Such information may include, for example, a category **610** and a name of the selected application **620** and/or a short text description **630** of the selected application.

Now referring to FIG. **7**, which shows a screenshot of user interface **56**, a user may select what error information **710** to display. A check mark **720** indicates that the corresponding error information will be displayed. The user may deselect particular error information by removing the corresponding check mark(s) **720**. Error information may relate to, for example, an identification number of an error, an identification of the application that reports an error, the username of the user who experiences an error, the date and time or an error, the data and time when an error was recorded, the name of a machine that experiences an error, an identification of the module in which an error occurs, and/or whether an error is recoverable. The error columns **730** shown in

FIG. **7** are defined in the module definition file that pertains to the module that the custom application belongs according to the module definition file. The column names shown in the first column of this view are suggested names and indicate to the user what data the column will show. In one or more embodiments, the user may rename a selected column to a user-defined column name by entering data in the column next to it (not shown). Such a renamed column will then be the name that is shown to the user when viewing the custom application.

Now referring to FIG. **8**, which shows a screenshot of user interface **58**, a user may select columns of data on which to perform operations. The user may select columns, specify an operation (e.g., add, multiply, subtract, divide, determine average, determine maximum, determine minimum) to perform on the selected columns, and have the results displayed in a new column.

Now referring to FIG. **9**, which shows a screenshot of user interface **60**, a user may select how to display selected columns of data. For example, the user may specify the number of records displayed on each page of a selected custom application, the number of records displayed by a search (of module data returned from a database, e.g., database **36** shown in FIG. **2**), a criteria for performing a quick search, a default criteria for sorting data, and/or a default sort direction (e.g., ascending, descending) of data returned by a search.

Further, in one or more embodiments, the user may rearrange columns in the order the user wishes to view them by dragging and dropping the column headers into any position in the view. Further still, in one or more embodiments, the user may group data by one or more columns by dragging and dropping the column header into the area above the view marked "Drag a column header here to group by that column". The alignment of each column may also be set by selecting the cell under the column header and clicking one of the "Cell Text Alignment" buttons marked "Left", "Center", and "Right" (not shown). This will result in the data being displayed with the selected justification when the user views the custom application.

Now referring to FIG. **10**, which shows a screenshot of user interface **62**, a user may select how to display and perform "footer" operations **1010**. Footer operations **1010** provide the ability to create custom footers for displaying calculations. Such footers are under the control of the user and may be displayed under the columns on which they perform calculations. Calculation results displayed in a footer may result from determining, for example, the average of the values in the column, the number of values in the column, the minimum value in the column, the maximum value in the column, and/or the sum of the values in the column.

Now referring to FIG. **11**, which shows a screenshot of user interface **64**, a user may select how displayed data is to be sorted. For example, the user may specify a criteria for searching data **1110**, a default criteria for sorting returned search data **1120**, and/or a default direction **1130** (e.g., ascending, descending) for sorting returned search data. In one or more embodiments, a drop-down menu may allow the user to select a column that can be "Quick Searched", meaning that the user will not have to popup a search window when viewing the custom application, but rather can enter search criteria at will for that column directly above the custom application view. Both the drop-down menus for "Quick Search" and "Sorting" may be defined in the module definition file.

Now referring to FIG. **12**, which shows a screenshot of user interface **66**, a user may create a new search (of module data returned from a database, e.g., database **36** shown in FIG. **2**). In creating a new search, the user may specify the type of search to conduct. The parameters available to the user for searching that are shown in the user interface **66** of FIG. **12** may be defined in the module definition file. Further, the type of data that the column pertains to may dictate what type of search criteria input the user may see. For example, if a column is defined as a "Date" type, the user may see a calendar date picker control, whereas if the type is "String", the user may see a textbox.

Now referring to FIG. **13**, which shows a screenshot of user interface **68**, a user may select which actions to perform on displayed data. For example, performable actions may include sorting data, searching data, and/or calculating values using displayed data. Further, now referring to a screenshot of user interface **70** shown in FIG. **14**, the user may select the order of actions to be performed on displayed data.

Further, in one or more embodiments, a user may be allowed to only generate custom displays of data. In other words, a user may not be allowed to perform actions on displayed data. The actions that are available to the user for choosing actions may be defined according to the module definition file.

Deleting Custom Applications

In one or more embodiments, a user may delete a custom application. This may be achieved by selecting a particular custom application from a list of available applications (e.g., the list of applications shown in FIG. **4**) and then deleting the selected custom application. Further, in one or more embodiments, only an owner of a custom application may be allowed to delete that custom application.

Copying Custom Applications

In one or more embodiments, a user may copy a custom application. This may be achieved by selecting a particular custom application from a list of available applications (e.g., the list of applications shown in FIG. **4**) and then copying the selected custom application as a new, possibly renamed, custom application. Such a feature allows the user to copy the attributes of a first custom application to a second custom application, thereby avoiding the need to wholly re-customize the second application.

Further, in one or more embodiments, a custom application may be shared among various users. Further still, in one or more embodiments, a custom application may be designated as "private", whereby users other than an owner of the custom application may be prevented from accessing the custom application. Further still, in one or more embodiments, an owner of a custom application may specify those users that may access the custom application. Further still, in one or more embodiments, an owner designation of a custom application may be changed either by the current owner or another user that has appropriate privileges.

Further, one or more embodiments of the present invention may be associated with virtually any type of computer system, including multiprocessor and multithreaded uniprocessor systems, regardless of the platform being used. For example, as shown in FIG. **15**, a networked computer system **200** includes at least one processor (e.g., a general-purpose processor, a field-programmable gate array (FPGA), an application-specific integrated circuit (ASIC), a graphics processor) **202**, associated memory **204**, a storage device **206**, and numerous other elements (not shown) and functionalities typical of modern computer systems. The networked computer system **200** may also include input means (e.g., a keyboard **208**, a mouse **210**) and output means (e.g.,

a monitor **212**). The networked computer system **200** may be connected to a local area network (LAN) or a wide area network (WAN) via a network interface connection (not shown). Those skilled in the art will appreciate that these input and output means may take other forms. Further, those skilled in the art will appreciate that one or more elements of the networked computer system **200** may be remotely located and connected to the other elements over a network. Further, software instructions to perform one or more embodiments of the present invention may be stored on a computer-readable medium such as a compact disc (CD), a diskette, a tape, a file, a hard drive, or any other computer-readable storage device.

Advantages of the present invention may include one or more of the following. In one or more embodiments, a user in supply chain management may select how and what supply chain data to view and/or perform actions on.

Further, in one or more embodiments, a user in supply chain management may customize supply chain management software applications to improve supply chain operations.

Further, in one or more embodiments, custom applications for supply chain management may be shared among multiple users.

While the invention has been described with respect to a limited number of embodiments, those skilled in the art, having benefit of the above description, will appreciate that other embodiments may be devised which do not depart from the scope of the invention as described herein. Accordingly, the scope of the present invention should be limited only by the appended claims.

What is claimed is:

1. A computer-implemented method for generating a custom application, comprising:

storing application parameters of a custom application for a trading partner in a supply chain network, the application parameters describing at least one user-customized display of supply chain data for the trading partner, the application parameters further defining a user-customized order of actions to be performed on the supply chain data and a type of the custom application;

receiving one or more user-modifications of the user-customized display of supply chain data and the user-customized order of actions to be performed on the supply chain data;

storing the received one or more user-modifications;

receiving a request for supply chain data over a computer network from the trading partner;

responsive to receiving the request for supply chain data from the trading partner, automatically identifying a role of the trading partner that made the request for supply chain data;

responsive to identifying the role of the trading partner, generating at least one of the application parameters based on the role of the trading partner;

generating a custom application comprising one or more user interfaces based on the stored application parameters and the at least one application parameter based on the role of the trading partner; and

rendering for display the custom application to the trading partner wherein the one or more user interfaces display portions of the requested supply chain data based on a format for display described in the application parameters and permit actions to be performed on the supply chain data based on the application parameters.

2. The computer-implemented method of claim **1**, further comprising:

receiving the application parameters as customized by the trading partner.

3. The computer-implemented method of claim **1**, wherein actions to be performed on the supply chain data comprise:

sorting data;

searching data; and

calculating values based on the supply chain data.

4. The computer-implemented method of claim **1**, wherein the user-customized display of supply chain data comprises supply chain data displayed in one or more columns that can be searched by a search feature comprising:

a drop-down menu permitting user selection of one or more columns;

an interface allowing a user to directly input search criteria for the one or more columns; and

the interface allowing the user to directly input search criteria comprises a calendar date picker when the selected one or more columns comprises a date type definition and comprises a textbox when the selected one or more columns comprises a string type definition.

5. A supply chain management system for generating a custom application, comprising:

a web interface tangibly embodied on a non-transitory computer-readable medium and configured to receive application parameters for a custom application, the application parameters describing at least one user-customized display of supply chain data, the application parameters further defining a user-customized order of actions to be performed on the supply chain data and a type of the custom application;

a custom application builder module tangibly embodied on the non-transitory computer-readable medium and configured to:

receive one or more user-modifications of the user-customized display of supply chain data and the user-customized order of actions to be performed on the supply chain data;

store the received one or more user-modifications;

receive a request for supply chain data over a computer network from a trading partner;

responsive to receiving the request for supply chain data from the trading partner, automatically identify a role of the trading partner that made the request for supply chain data;

responsive to identifying the role of the trading partner, generate at least the one application parameter based on the role of the trading partner;

generate a custom application comprising one or more user interfaces based on the application parameters and the at least one application parameter based on the role of the trading partner; and

render for display the custom application to the trading partner wherein the one or more user interfaces display portions of the requested supply chain data based on a format for display described in the application parameters and permit actions to be performed on the supply chain data based on the application parameters.

6. The supply chain management system of claim **5**, wherein the custom application builder module is further configured to cause the web interface to render a display of at least a portion of the supply chain data returned from a database in response to the request.

7. The supply chain management system of claim **5**, wherein the application parameters describe at least one customized display of the custom application.

8. A computer-implemented method, comprising:

selecting a type of application that displays supply chain data, the type of application related to a user-created application and defining application parameters, the supply chain data stored in a database;

receiving one or more user-modifications of a user-customized display of the supply chain data from a trading partner;

storing the received one or more user-modifications;

generating a web-based request for the supply chain data based on the selecting;

selectively customizing a display of at least a portion of the supply chain data returned from the database in response to the web-based request;

generating at least one application parameter based on a role of the trading partner; and

generating a custom application comprising one or more user interfaces based on the application parameters and the at least one application parameter based on the role of the trading partner and rendering for display the customized display to the trading partner based on a format for display described in the at least one application parameter and permit actions to be performed on the supply chain data based on the at least one application parameter.

9. The computer-implemented method of claim 8, the act of selectively customizing comprising at least one of:

selecting for display at least one column comprising at least the portion of the supply chain data;

selecting a name of the at least one column;

selecting a criteria for sorting data in the at least one column;

selecting a direction for sorting data in the at least one column;

selecting a search criteria for searching the returned supply chain data;

defining the search criteria for searching the returned supply chain data;

selecting error information to display;

selecting an action for performing at least one operation on data in the at least one column; and

naming the action.

10. The computer-implemented method of claim 8, further comprising:

performing an operation on data in a displayed column comprising at least the portion of the returned supply chain data.

11. The computer-implemented method of claim 10, the act of performing comprising at least one of:

adding the data in the displayed column;

averaging the data in the displayed column;

determining a maximum value in the displayed column;

determining a minimum value in the displayed column;

performing an operation on the data in the displayed column with data in another displayed column comprising at least the portion of the returned supply chain data; and

performing the operation on the data in the displayed column with data in another displayed column comprising at least the portion of the returned supply chain data and forming a new displayed column based on a result of the performing the operation on the data in the displayed column with the data in the another displayed column.

12. The computer-implemented method of claim 10, wherein data displayed in one or more columns can be searched by a search feature comprising:

a drop-down menu permitting user selection of one or more columns;

an interface allowing a user to directly input search criteria for the one or more columns; and

the interface allowing the user to directly input search criteria comprises a calendar date picker when the selected one or more columns comprises a date type definition and comprises a textbox when the selected one or more columns comprises a string type definition.

13. The computer-implemented method of claim 10, wherein footer data is displayed comprising one or more formulas that generate the data displayed in the column.

14. A supply chain management system, comprising:

a computer system providing a user a capability to build a custom application requiring supply chain data;

a user interface tangibly embodied on a non-transitory computer-readable medium that receives one or more user-modifications of a user-customized display of the supply chain data and a user-customized order of actions to be performed on the supply chain data;

a web interface tangibly embodied on the non-transitory computer-readable medium and accessible by the computer system, wherein the web interface handles a request from a computer for the supply chain data; and

a database that stores the received one or more user-modifications of the user-customized display of the supply chain data, wherein a procedure in metadata stored in the database is invocable to return the supply chain data based on parameters in the request, the parameters comprising the user-customized order of actions to be performed on the supply chain data and a type of the custom application, wherein the computer system generates the custom application based on the parameters and a role of a trading partner and renders for display the user-customized display to the trading partner based on a format for display described in at least one of the parameters and permits actions to be performed on the supply chain data based on the at least one of the parameters.

15. The supply chain management system of claim 14, wherein the parameters in the request are specific to the type of the custom application.

16. The supply chain management system of claim 14, wherein the metadata comprises an XML file.

17. The supply chain management system of claim 14, wherein the web interface is further configured to authenticate the user.

18. The supply chain management system of claim 14, wherein at least a portion of the returned supply chain data is displayed to the user using the custom application, wherein the user is capable of selecting display preferences for the custom application.

19. The supply chain management system of claim 14, wherein supply chain data stored in the database is updateable based on activities in a supply chain related to the user.

20. The supply chain management system of claim 14, wherein the database is configured to store a plurality of metadata, each of which corresponds to a module associated with the type of custom application selectable by the user.

21. A non-transitory computer-readable medium embodied with software for generating a custom application, the software when executed using a computer system is configured to:

render a web-based application configured to receive a selection of a first application to customize, the first

application having a type that is related to a user-created application, the first application requiring supply chain data;

receive one or more user-modifications of a user-customized display of the supply chain data and a user-customized order of actions to be performed on the supply chain data;

store the received one or more user-modifications;

receive a request for supply chain data over a computer network from a trading partner;

responsive to receiving the request for supply chain data from the trading partner, automatically identify a role of the trading partner;

responsive to identifying the role of the trading partner, generate a custom application comprising one or more user interfaces based on the user-customized display of the supply chain data and the user-customized order of actions to be performed on the supply chain data and the role of the trading partner; and

render for display the custom application to the trading partner wherein the one or more user interfaces display portions of the requested supply chain data based on a format for display and permit actions to be performed on the supply chain data based on the user-customized display of the supply chain data and the user-customized order of actions to be performed on the supply chain data and the role of the trading partner.

22. The non-transitory computer-readable medium of claim **21**, wherein the software is further configured to:

allow a user to update supply chain data stored in a database.

23. The non-transitory computer-readable medium of claim **21**, wherein the software is further configured to:

allow a user to at least one of delete, copy, modify, restrict, and share with another user the custom application.

* * * * *